

第7章

転移学習の導入による Random Forest の効率化

本章では、転移学習をマルチクラス分類問題へ展開する。ここでは、マルチクラス識別器として様々な応用に利用されている Random Forest を対象とし、転移学習の導入法について述べる。まず、Random Forest の概要について述べた後、転移学習の Random Forest への導入について述べる。Random Forest において転移学習を実現することで、マルチクラスの物体検出問題においても効率的な識別器の構築が期待できる。

7.1 マルチクラス分類における転移学習

5章と6章で示した学習サンプルの生成や転移学習を用いることで、2クラス識別器の再学習を大きく効率化することができた。しかし、2000年代以降の物体検出では1クラスのみでなく、画像中に存在する複数の物体クラスを認識するマルチクラス分類問題も重要なタスクとされている [41][42]。マルチクラス分類問題で注目されている識別器に Random Forest がある。Random Forest は高い汎化性能を持つ上に、決定木の集合体であることから並列処理による高速な識別が可能である。しかし、Random Forest もまた統計的学習法であるため、その識別性能は学習サンプルに依存し、設置環境の変化に対する再学習は避けられない。そこで、Random Forest に転移学習を導入することで、マルチクラス物体検出に対しても識別器の構築における効率化が可能となる。本章では、マルチクラス分類問題における設置環境への対応の効率化について述べる。

7.2 Random Forest

Random Forest [43] とは、2001年に Breiman によって提案された複数の決定木構造を持ったマルチクラス識別器を構築するアンサンブル学習アルゴリズムである。図 7.1 に Random Forest の構造を示す。このアルゴリズムの特徴は、Bagging [44] と同様にブートストラップを取り入れ過学習を防ぐことが可能な点や、Random Feature Selection を取り入れることで特徴ベクトルの次元数が大きくても高速に学習が可能である点である。このようなメリットを持ち、クラス識別、クラスタリング、回帰分析に利用することができるため、様々な分野に応用されている。

コンピュータビジョンの分野でも、文字認識 [45]、物体認識 [46]、物体セグメンテーション [47]、人体姿勢推定 [48]、医用画像処理 [49] で用いられており、それぞれの研究課題のブレイクスルーを実現するフレキシブルな学習アルゴリズムとして利用されている。

7.2.1 Random Forest の学習アルゴリズム

Random Forest は、学習サンプルを用いて図 7.1 に示すような複数の決定木構造を持つ識別器を構築する。各決定木は、分岐ノード (Split Node) と末端ノード (Leaf Node) により構成され、分岐ノードを繰り返し作成し、一定の基準により分岐が不可能になった際に、末端ノードを作成することにより構築される。分岐ノードは、ある 1次元の特徴量としきい値の大小関係により左右に分岐する。また、末端ノードは辿り着いた学習サンプルの各クラス確率を保存する。これらの構造を持つ識別器の構築アルゴリズムについて以下に述べる。

Random Forest の学習アルゴリズムを **アルゴリズム 7.1** に示す。まず、学習サンプルを I 、木の数 T 、木の深さ D 、サブセット $I = (I_1, I_2, \dots, I_T)$ 特徴選択回数 F 、しきい値選択回数 TH を定義する。 F, TH は分岐ノードを作成するためのパラメータである。サブセットは学習サンプルからランダムで選択した小さなデータ集合であり、サブセット間のデータのオーバーラップを許容して作成される。これらのサブセットを利用することにより決定木を構築する。

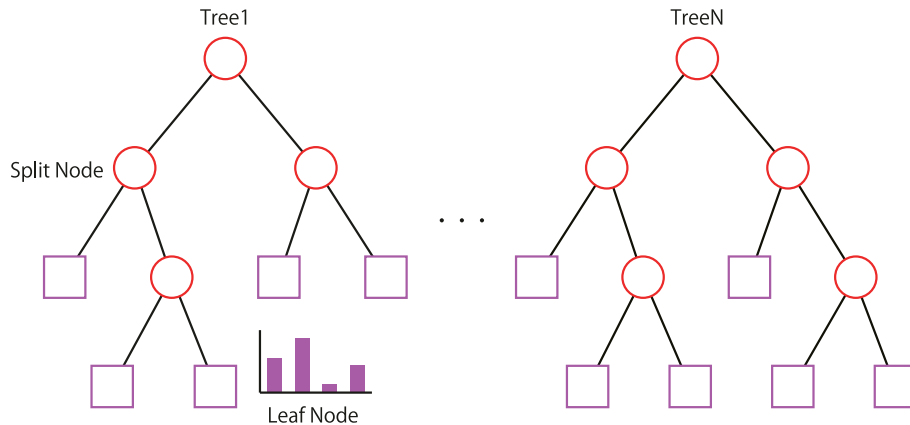


図 7.1: Random Forest の構造.

決定木の構築は、分岐が不可能になるまで分岐ノードを繰り返し作成する。分岐ノードはある特徴量 f としきい値 t を用いて左もしくは右への分岐を決めるものである。そのため、 f と t は様々な候補の中から厳選する必要がある。しかし、これらの候補を全探索すると特徴次元が大きくなるに従い学習に時間を要するという問題が発生する。この問題を解決し、高速な学習を実現するために Random Feature Selection を導入する。予め定めた特徴選択回数 F としきい値選択回数 TH を用いて、分岐ノードの候補を $F \sim TH$ の組み合わせだけ用意する。この候補の中のみから最適なものを探すことにより、高速に学習を行う。最適な分岐ノードを探す基準に情報利得 ΔE を利用し、 ΔE が最も大きくなる組み合わせを分岐ノードとして用いる。ある分岐ノード n に分かれてきたデータ集合 I_n 、分岐ノード組み合わせ f と t を用いて式 (7.1)、式 (7.2) により I_l と I_r を算出する。

$$I_l = \{i \in I_n \mid f(\mathbf{v}_i) < t\} \quad (7.1)$$

$$I_r = I_n \setminus I_l \quad (7.2)$$

この I_l と I_r を用いて、式 (7.3) により情報利得 ΔE を算出する。

$$\Delta E = E(I) - \frac{|I_l|}{|I_n|} E(I_l) - \frac{|I_r|}{|I_n|} E(I_r) \quad (7.3)$$

ここで、関数 $E(I)$ は式 (7.4)(情報エントロピー)、または式 (7.5)(ジニ係数) である。

$$E(I) = - \sum_{i=1}^n p(c_i) \log p(c_i) \quad (7.4)$$

$$E(I) = \sum_{i=1}^n p(c_i) (1 - p(c_i)) \quad (7.5)$$

ここで、 $p(c_i)$ はクラス c_i の確率 (相対頻度) を表しており、学習サンプルにつけられている教師信号により求められる。これらの処理を繰り返すことにより分岐ノードを作成してゆき、情報利得が 0

アルゴリズム 7.1 : Random Forest の学習アルゴリズム

1. 入力: 学習サンプル : I を用意する.

パラメータ: 木の数 : T , 木の深さ: D

2. サブセット作成: T 個のサブセット : $I = (I_1, I_2, \dots, I_T)$

For $k = 1, \dots$, 木の数 T まで

サブセット I_k を用いて決定木を作成

For $l = 1, \dots$, 特徴量選択数 F まで

分岐関数 f をランダムに選択

For $m = 1, \dots$, しきい値選択数 TH まで

×しきい値 t をランダムに選択

× f と t を用いてデータを I_l と I_r に分岐

$$I_l = \{i \mid I_n \setminus f(\mathbf{v}_i) < t\}$$

$$I_r = I_n \setminus I_l$$

×教師信号を利用して情報利得 ΔE を算出

$$\Delta E = \frac{|I_l|}{|I_n|} E(I_l) - \frac{|I_r|}{|I_n|} E(I_r)$$

×if $\Delta E > \Delta E_{old}$ f, t, I_l, I_r を保存

End for

End for

If $gain = 0$ または D に達した \rightarrow 末端ノードを作成

教師信号を用いて $P(c|\mathcal{V})$ を計算

×Else I_l, I_r を用いて再帰処理

End for

3. 出力: 識別器

未知サンプル \mathbf{x} が到達したノード l よりクラス c の確率を算出

$$P(c|\mathbf{x}) = \frac{1}{T} \sum_{t=1}^T P_t(c|\mathcal{V})$$

クラス確率が最大のクラスに属すると推定

$$C_i^* = \arg \max_{c_i} P(c_i|\mathbf{x})$$

になった場合に末端ノード l を作成し, 分岐されてきたデータを用いて確率 $P(c|\mathcal{V})$ を計算する. これにより, 各決定木の学習が可能となる.

7.2.2 Random Forest の識別処理

Random Forest の識別処理の流れを、図 7.2 に示す。未知データを \boldsymbol{x} とした時に各決定木に入力し、たどり着いた末端ノードに保存されている $P(c|\boldsymbol{x})$ を出力する。各決定木から出力された $(P_1(c|\boldsymbol{x}), P_2(c|\boldsymbol{x}), \dots, P_T(c|\boldsymbol{x}))$ を用いて以下の式により未知データの事後確率 $P(c|\boldsymbol{x})$ を算出する。

$$P(c|\boldsymbol{x}) = \frac{1}{T} \sum_{t=1}^T P_t(c|\boldsymbol{x}) \quad (7.6)$$

これを用いて次式の単純ベイズ識別器により未知データのクラスを推定する。

$$C_i^* = \arg \max_{c_i} P(c_i|\boldsymbol{x}) \quad (7.7)$$

以上の処理によりマルチクラス識別が可能となる。

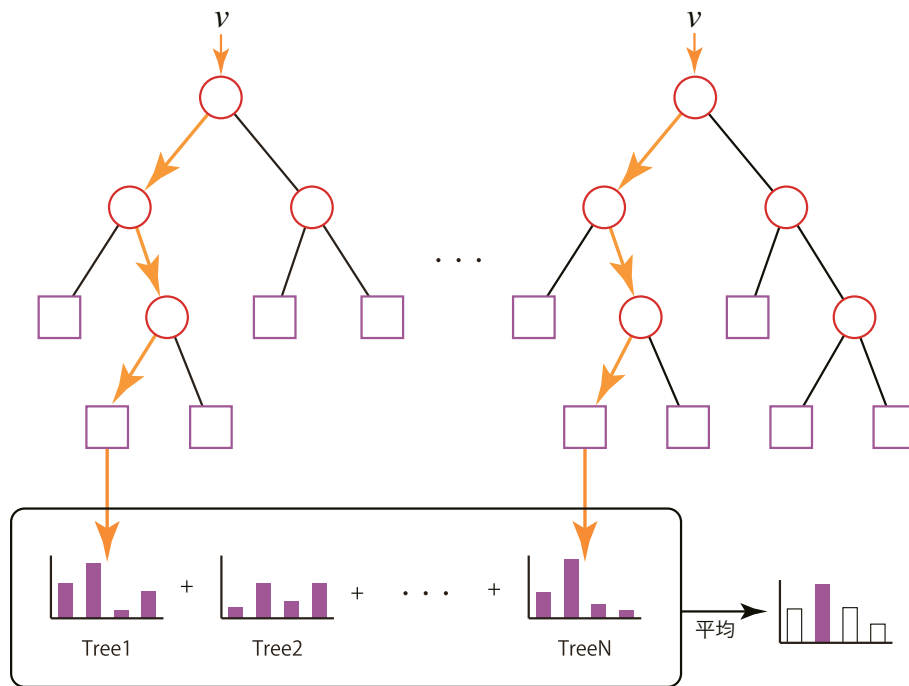


図 7.2: Random Forest による識別。

7.3 Random Forestの転移学習

Random Forest はアンサンブル学習を容易に実現できることから、様々な問題に展開されている。しかしながら、統計的学習法である以上、学習環境と設置環境が大きく異なる際の性能低下を避けられず、再学習を必要とする。この再学習に転移学習を導入することで識別器の構築における効率化を実現する。本節では共変量シフトに基づく転移学習を導入した Random Forest のアルゴリズムについて述べる。

7.3.1 共変量シフトに基づく転移学習を導入した Random Forest

共変量シフトに基づく転移学習を導入した Random Forest の学習の流れを図 7.3 に示す。

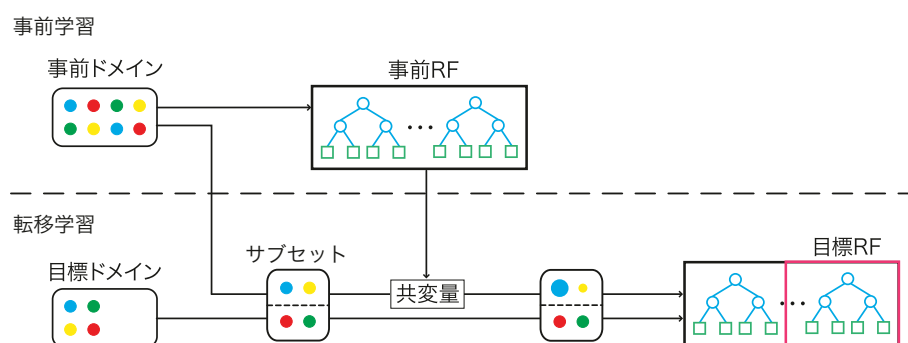


図 7.3: 共変量シフトに基づく転移学習を導入した Random Forest.

転移学習の前提として、十分な事前ドメインの学習サンプルが存在し、その学習サンプルを用いた Random Forest は構築済みであるとする。この事前ドメイン $X_a = \{x_1, x_2, \dots, x_j\}$ と事前に構築した Random Forest (事前 RF) を、少数用意した目的ドメインの学習サンプル $X_t = \{x_1, x_2, \dots, x_i\}$ の学習に利用する。この際、事前ドメインサンプルは共変量シフトにより目標ドメインの学習への有効性で重み付けされ、有効でない事前ドメインサンプルは、転移学習ではその影響を失う。このとき共変量は、事前 RF と現在構築している目標 Random Forest (目標 RF) の出力の比により決定する。

決定木の転移学習は、図 7.4 に示すように以下の 3 つのステップで行われる。

- 1) 事前ドメイン X_a 、目標ドメイン X_t から同数のサンプルを抽出し、サブセットを作成する。
- 2) サブセットに対し決定木を学習する。ここで、サブセット内の X_a に属するサンプルは共変量 λ で重み付けされている。
- 3) 構築した決定木を目標 Random Forest に加え更新する。更新された目標 Random Forest と事前学習した Random Forest により、共変量 λ を更新する。

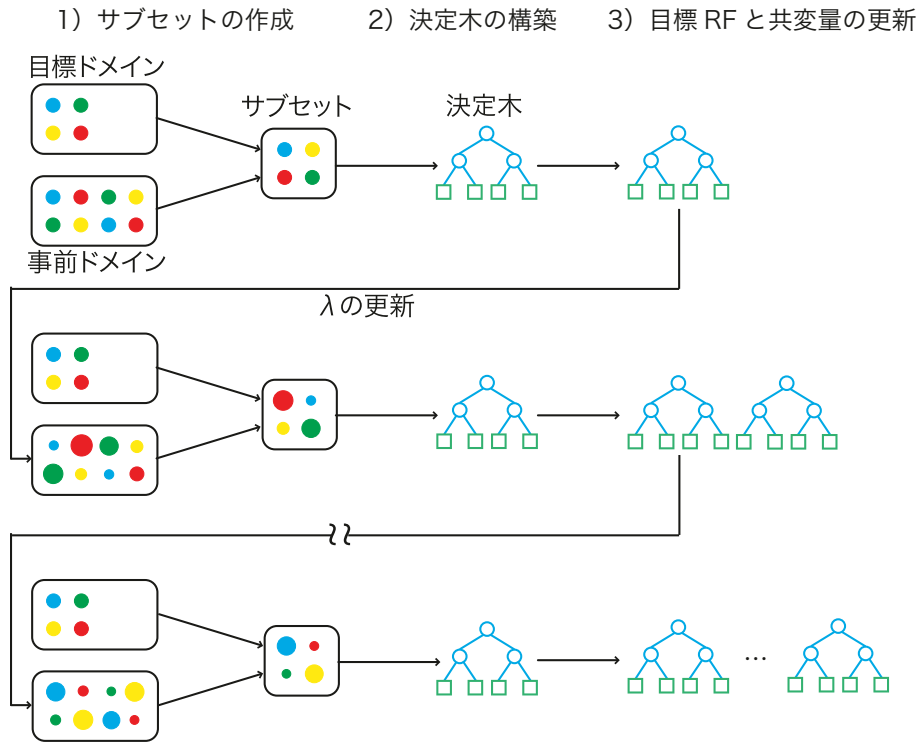


図 7.4: 転移学習を導入した Random Forest の構築.

以上のステップを繰り返す、十分な数の決定木となるまで目標ドメインに対する Random Forest を構築する。通常の Random Forest との違いは、入力サンプルが部分的に重み付けられている点と、その重みを更新する点である。ここで、各決定木に与えられるサブセット I は、 X_a と X_t から同数サンプルを採用することで作成される。これにより、事前ドメインと目標ドメインの数量差を無視できる。共変量 λ は 6 章で述べた式 (6.6) の、Boosting による識別器の出力 H を、式 (7.6) で示した Random Forest の出力 $P(c|\mathbf{x})$ に置き換えた次式で求める。

$$\lambda_j = \frac{1 + e^{P_a(c_k|\mathcal{X}_j)}}{1 + e^{P_t(c_k|\mathcal{X}_j)}} \quad (7.8)$$

ここで $P_a(c_k|\mathcal{X}_j)$ は j 番目の事前ドメインサンプル \mathcal{X}_j を事前 Random Forest に入力した際の正解クラスラベル c_k の尤度、 $P_t(c_k|\mathcal{X}_j)$ は同様に構築中の目標 Random Forest に入力した際の尤度である。一本目の決定木を構築する際は、 $P_t(c_k|\mathcal{X}_j)$ は 0 とし、 λ_j を $1 + e^{P_a(c_k|\mathcal{X}_j)}$ で初期化する。

分岐関数の決定は、通常の Random Forest 同様にランダムに選択された特徴量としきい値に対し

て情報利得を算出し、情報利得が最大となる候補を分岐関数として採用する。

$$\Delta E = \frac{|I_l|}{|I|} E(I_l) - \frac{|I_r|}{|I|} E(I_r) \quad (7.9)$$

$$E(I) = \sum_{k=1}^C P(c_k) \log P(c_k) \quad (7.10)$$

提案手法は共変量を用いて、目的ドメインの学習に有効でない事前ドメインサンプルの影響が小さくなるように事前ドメインサンプルを重み付ける。そのため、次式のようにドメインごとに異なる計算でクラス確率を算出して統合することで、両ドメインのクラス確率 $P(c_k)$ を算出する。

$$P(c_k) = \frac{|I^{t,c_k}| + |I^{a,c_k}|}{|I^t| + |I^a|} \quad (7.11)$$

$$|I^{a,c_k}| = \sum_{j:y_j=c_k} \lambda_j \quad (7.12)$$

ここで、 $|I^{t,c_k}|$ はサブセット内の目標ドメインにおいてクラスラベルが c_k であるサンプルの数、 $|I^{a,c_k}|$ は事前ドメインにおいてクラスラベルが c_k であるサンプルに付与された λ の総和である。また、 $|I^t|$ は全てのクラスについて $|I^{t,c_k}|$ を総和したものであり、 $|I^a|$ は全てのクラスについて $|I^{a,c_k}|$ を総和したものである。

任意の深さに達するまで分岐を繰り返し決定木を構築した後、 λ を式 (7.8) に基づき更新し、次の新しい決定木を作成していく。決定木の作成は、決定木の本数が任意の数に達することで終了する。以上の処理により、任意の本数の決定木で構成される Random Forest を構築することができる。ここで、転移学習における事前ドメインサンプルの選択基準である λ は、初期状態では 1 が与えられ、決定木が追加される度に目標 RF と事前 RF の違いを正しく表現できるよう更新される。そのため、学習の序盤に構築された決定木における λ の信頼性は低く、目標ドメインの学習に適合していない可能性がある。Boosting に基づく転移学習の 1 つである TrAdaBoost[38] は、同様の理由で T ラウンド学習した際の前半部分は切り捨て、後半 $T/2$ の弱識別器のみで識別を行っている。本研究では文献 [38] と同様に、前半 $T/2$ の決定木を切り捨てることで、学習初期の悪影響を低減する。

提案手法の学習アルゴリズムを **アルゴリズム 7.2** に示す。

アルゴリズム 7.2: 共変量シフトによる転移学習を導入した Random Forest の学習

1. 入力: I 個の目標ドメイン学習サンプル X_t と J 個の事前学習サンプル X_a とそれらのクラスラベル c_k を用意する.

事前に X_a を Random Forest により学習し, 識別器を構築しておく.

2. 初期化: 事前ドメインの重み付けとなる共変量 λ_j を初期化

$$\lambda_j = 1 + e^{P_a(c_k|\mathcal{X}_j)}$$

3. 学習:

For $t = 1, \dots, T$ //決定木の本数

\mathcal{X}_t, X_a から同数のサンプルを無作為に選択し, サブセットを作成

 決定木の作成 (アルゴリズム 7.1)

 クラス確率

$$P(c_k) = \frac{|\mathcal{I}^{t,c_k}| + |\mathcal{A}^{a,c_k}|}{|\mathcal{I}^t| + |\mathcal{A}^a|}$$
$$|\mathcal{A}^{a,c_k}| = \sum_{j: y_j = c_k} \lambda_j$$

λ_j の更新

$$\lambda_j = \frac{1 + e^{P_a(c_k|\mathcal{X}_j)}}{1 + e^{P_t(c_k|\mathcal{X}_j)}}$$

End for

4. 出力: 目標ドメインに対する Random Forest

後半部分 ($t = T/2$) の決定木のみを抽出し, Random Forest とする.

$$P(c_k|\mathbf{x}) = \frac{2}{T} \sum_{t=T/2}^T P(c_k|\mathcal{V})$$

7.4 評価実験

転移学習を導入した Random Forest の有効性を評価するため、目標ドメインに事前ドメインを加えたデータ集合を用いて再学習した Random Forest と目標ドメインのみで再学習した Random Forest と比較する。

7.4.1 2クラス分類問題の転移

2クラス分類問題での評価実験は、事前ドメイン、目標ドメインの評価データとして、異なる人検出データセットを用いる。本実験では、事前ドメインとして DaimlerChrysler dataset を、目標ドメインとして INRIA person dataset を用いる。各ドメインの例を図 7.5 に示す。



(a)DaimlerChrysler Mono Pedestrian Detection Benchmark Dataset



(b)INRIA Person Dataset

図 7.5: 各データベースのサンプル例。

事前ドメイン：DaimlerChrysler Mono Pedestrian Detection Benchmark Dataset[50]

15,600 枚の人画像と 6,700 枚の背景画像が学習サンプルとして、56,500 の人領域を含む 21,800 枚の画像が評価サンプルとして用意されている。

目標ドメイン：INRIA Person Dataset[3]

2,416 枚の人画像と 1,218 枚の背景画像が学習サンプルとして、1,135 の人画像と 453 枚の背景画像が評価サンプルとして用意されている。

入力特徴量は HOG 特徴量 [3] とする。Random Forest 構築の際のパラメータを 7.1 に示す。

表 7.1: Random Forest 構築のパラメータ.

木の本数	50
木の最大深さ	5
特徴次元数	3,780
分岐ノード候補のランダム生成数	$63(≐\sqrt{3,780})$

事前ドメインに目標ドメインを転移する提案手法と以下の2つの手法を比較する.

“事前ドメイン+目標ドメイン”

事前ドメインに目標ドメインを転移を用いず加えたサンプル集合を入力として Random Forest を学習する.

“目標ドメインのみ”

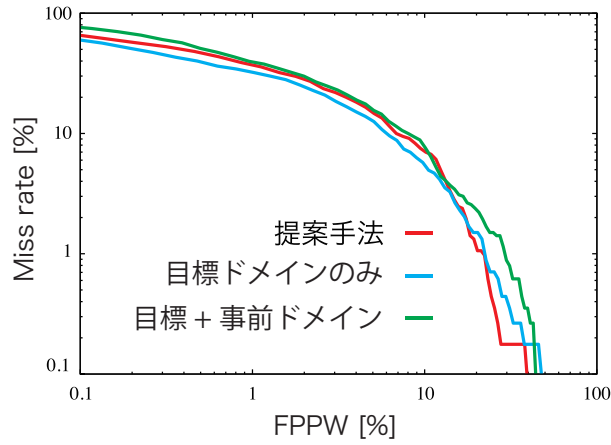
目標ドメインのサンプルのみを入力として Random Forest を学習する.

7.4.2 2クラス分類問題への転移学習の導入による効率化

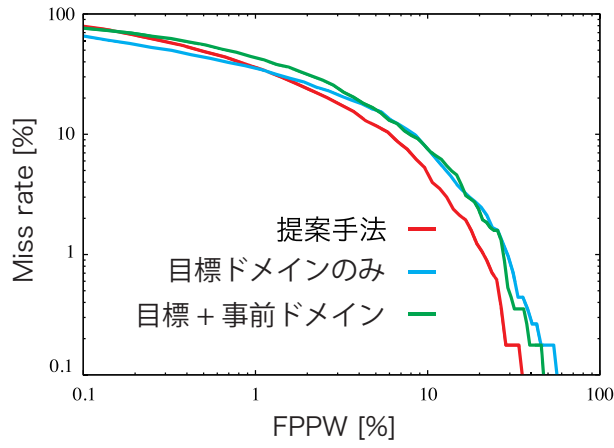
目標ドメインのポジティブ学習サンプルを 2,416 枚から 800 枚, 100 枚と順に削減した際の DET カーブを図 7.6 に示す. まず, 図 7.6(a) より, 目標ドメインの学習サンプルを 2,416 枚全て使用した場合, 提案手法と “事前ドメイン+目標ドメイン”, “目標ドメインのみ” は同程度の性能となることがわかる.

800 枚に学習サンプルを削減した場合, 図 7.6(b) のように “目標ドメインのみ”, “事前ドメイン+目標ドメイン” は性能を低下させる. これは, 目標ドメインの学習サンプル数が削減されることで目標ドメインに対する学習が十分でなくなるためである. 一方, 提案手法はその性能を維持していることがわかる.

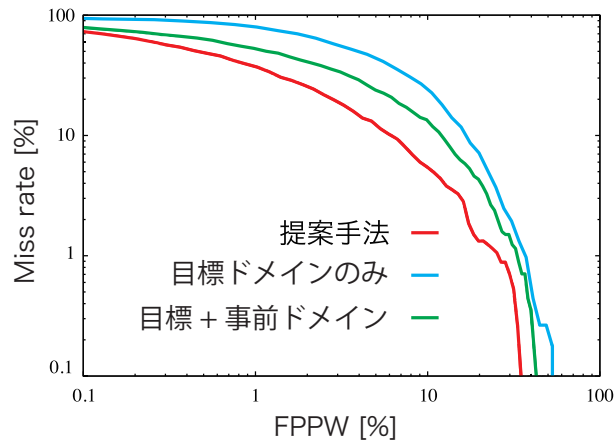
さらに, 目標ドメインの学習サンプルを 100 枚まで削減した場合, 図 7.6(c) より, “目標ドメインのみ” の識別性能は大きく低下する. これは, 100 枚の人画像の学習ではデータセット全体の分布を捉えることはできないことを示している. また, “事前ドメイン+目標ドメイン” は性能の低下が緩やかである. これは, 事前ドメイン中に存在する目標ドメインの識別に有効なサンプルを学習できたためであると考えられる. しかし, 事前ドメイン中には目標ドメインの学習に適さないサンプルも多く含まれるため, 2,416 枚や 800 枚での結果に比べ識別性能は低下している. 一方, 提案手法は高い識別性能を獲得しており, その識別性能は 2,416 枚, 800 枚時と比較しても大きく低下しない. 以上より, 転移学習により識別に有効な事前ドメインサンプルのみを重視しながら, 目標ドメインの学習に取り込むことで, 目標ドメインの学習サンプルの不足を補うことができたと考えられる.



(a) 入力サンプル 2,416 枚



(b) 入力サンプル 800 枚



(c) 入力サンプル 100 枚

図 7.6: 転移学習を導入した Random Forsts の識別性能.

7.4.3 マルチクラス分類問題の転移

マルチクラス分類問題での評価実験では、人体モデルの向きを示すパラメータ θ を正面、右向き、背面、左向きと 90° ずつ 4 分割した生成人体画像データを用いる。ここで、正面と背面を同一クラスとし、正面／背面、右向き、左向きの 3 クラスに背景クラスを加えた計 4 クラスをカテゴリとする。

本実験では、事前ドメインとして俯角 20° の生成人体画像データセットを、目標ドメインとして俯角 30° の生成人体画像データセットをそれぞれ用いる。各データセットは 5 章で述べた学習サンプル生成手法により生成した。図 7.7 に、各ドメイン、各クラスのサンプル画像例を示す。

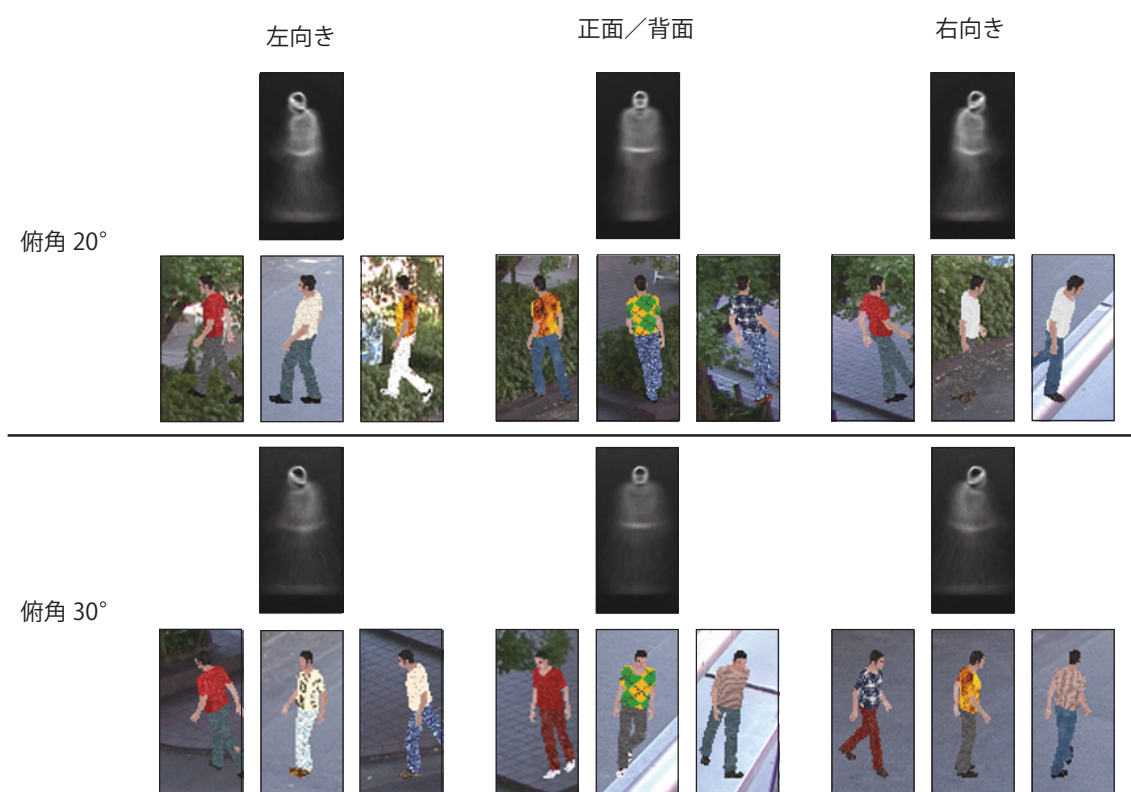


図 7.7: マルチクラス分類問題のサンプル例。

事前ドメイン：俯角 20°

俯角を 20° に設定し、2,416 枚の人画像と 12,180 枚の背景切出し画像を学習サンプルとして生成する。

目標ドメイン：俯角 30°

俯角を 30° に設定し、2,416 枚の人画像と 12,180 枚の背景切出し画像を学習サンプルとして、同数の画像を評価サンプルとして用意する。俯角の変化により、各カテゴリの見えは大きく変化する。図 7.7 の俯角 20° と俯角 30° の平均匀配画像を比較すると、同じ左向きでも体の角度がより広く拡散

し、ぼやけていることがわかる。

7.4.4 マルチクラス分類問題への転移学習の導入による効率化

目標ドメインのポジティブ学習サンプルを 2,416 枚, 800 枚, 100 枚と削減した際の識別性能を, それぞれ表 7.2, 表 7.3, 表 7.4 に示す。

表 7.2: マルチクラス分類問題に対する転移学習結果 (2,416 枚)。

	正面／背面 [%]	右向き [%]	左向き [%]	背景 [%]	平均 [%]
提案手法	55.8	91.0	88.1	97.1	83.0
目標ドメインのみ	60.0	90.8	88.7	98.5	84.5
事前ドメイン+目標ドメイン	53.5	87.6	84.6	98.8	81.1

表 7.3: マルチクラス分類問題に対する転移学習結果 (800 枚)。

	正面／背面 [%]	右向き [%]	左向き [%]	背景 [%]	平均 [%]
提案手法	55.0	77.5	73.7	98.9	76.3
目標ドメインのみ	60.4	81.5	77.8	98.8	79.6
事前ドメイン+目標ドメイン	42.9	77.8	73.2	94.6	72.1

表 7.4: マルチクラス分類問題に対する転移学習結果 (100 枚)。

	正面／背面 [%]	右向き [%]	左向き [%]	背景 [%]	平均 [%]
提案手法	44.4	70.9	71.9	98.8	71.5
目標ドメインのみ	47.6	28.8	43.7	98.9	54.8
事前ドメイン+目標ドメイン	36.1	65.3	67.1	94.7	65.8

表 7.2 より, 目標ドメインの学習サンプルを 2,416 枚全てを使用した場合, 提案手法, “目標ドメインのみ”, “事前ドメイン+目標ドメイン” の 3 手法ともほぼ同程度の平均性能となることがわかる。

次に, 800 枚に学習サンプルを削減した場合, 表 7.3 より, 各手法とも精度が低下しており, サンプル数が不足していることがわかる。特に, “事前ドメイン+目標ドメイン” は 72.1% と大きく性能を低下させている。これは, マルチクラス分類問題において俯角が異なると, 事前ドメインと目標ドメインの差が大きくなり, 適応できない事前ドメインサンプルが多く発生するためであると考えられる。提案手法は転移学習により, 適応できない事前ドメインサンプルを学習に用いないことで, “事前ドメイン+目標ドメイン” に比べ性能低下を抑制できたと考えられる。

さらに, 目標ドメインの学習サンプルを 100 枚まで削減した場合, 表 7.4 より, “目標ドメインのみ” の識別性能は大きく低下する。特に, 右向き, 左向きについての識別率が顕著に低下しており,

両カテゴリについて学習サンプルが十分でないことがわかる。また，“事前ドメイン+目標ドメイン”と提案手法は性能の低下が緩やかである。これは、目標ドメインで不足している右向きカテゴリと左向きカテゴリに有効なサンプルを、事前ドメインから補い学習できたためであると考えられる。しかし、事前ドメイン中には目標ドメインの学習に適さないサンプルも多く含まれるため、共変量を用いた転移学習を行う提案手法が高精度である。

7.5 まとめ

本章では、Random Forest に転移学習を導入することによる再学習の効率化について述べた。決定木構築時に共変量シフトを用いることで、目標ドメインの学習に適した事前ドメインサンプルのみを利用して学習できる。このため、少数の目標学習サンプル追加で十分な学習が可能となる。評価実験において、目標ドメイン数を 100 枚と少量にした際に、目標ドメインのみの再学習では著しく性能が低下するのに対して、提案手法は学習サンプルを $800 \in 2,416$ 枚使用した際と同程度の識別性能を得ることができた。これは、多くのカテゴリについて大量にサンプルを得ることが困難なマルチクラス分類問題において有効である。これにより、提案手法は Random Forest をベースとした様々なアプリケーションにおける学習の効率化に利用が可能である。

