卒 業 論 文 概 要 書

Summary of Bachelor's Thesis

Date of submission: 02/07/2020

専攻名（専門分野） Department	情報通信	氏名 Name	安藤由莉	指　導 教　員 Advisor	渡辺 裕　　印
研究指導名 Research guidance	オーディオビジュアル 情報処理研究	学籍番号 Student ID number	1w162020-7 CD		
研究題目 Title	画像特徴量を用いたファッションアイテム推薦システムの検討 A Study on Fashion Item Recommendation System Using Image Features				

1. まえがき

近年，ファッション電子商取引業界の市場規模は急速に拡大している．しかし，実店舗で好みの衣服を探すのに比べ，インターネット上で欲しいアイテムを見つけることは，アイテムの数が膨大であることなどから考えて困難である．こうした背景から，これまで一人ひとりの趣味嗜好に合わせたファッションアイテムの推薦に関する研究が多く行われてきた．

ユーザーの過去のデータなどを推薦に用いるとどうしても莫大なデータが必要になってしまう．そこで，商品画像のみから推薦するアイテムを選択するコンテンツベースの手法をとる．また，ファッションの重要な要素である色と形という点に注目することで，よりファッションアイテムの推薦として適切なものに近づくと考えられる．本研究では，新しく好みのファッションアイテムを選択するとそのアイテムに類似したアイテムを推薦するシステムを提案する．

2. 従来手法とその問題点

視覚的特徴を考慮したファッションアイテムの推薦システムに関する先行研究に，Gorripati らの手法がある．Gorripati らは，CNN の全結合層の特徴量を取り出し，類似度判定を行い推薦するアイテムを提示している [1]．しかし，この手法では色や形を考慮した推薦がされているとは言えない．

3. 提案手法

Deepfashion2 をデータセットに用いる [2]．画像の前処理として，セグメンテーションと色分類を行った．

データセットを表 1 のようにカテゴリ分類し，大カテゴリごとに 2 通りの学習を行った．手法 1 では，アイテムを小カテゴリで分類し，さらに色分類したクラスを出力層として，セグメンテーション後の画像を用いて VGG16 モデル[3]を fine tuning して学習する．手法 2 はセグメンテーション前の画像を用いて，アイテムを小カテゴリで分類したクラスを出力層として VGG16 モデルを fine tuning して学習する．

学習終了後，すべての画像を手法 1 と手法 2 の学習モデルに通す．出力層の直前の全結合層の値を取り出し，大カテゴリごとのファイルに保存する．そしてある一つのアイテムを選んだ時，そのアイテムと同じ大カテゴリのすべてのアイテムについて特徴ベクトルのコサイン類似度計算を行い，最も類似度が高いものから順に表示する．

表 1　データセットのカテゴリ分け

大カテゴリ	小カテゴリ
tops	short sleeve top, long sleeve top, vest, sling
bottoms	shorts, trousers, skirt
dress	short sleeve dress, long sleeve dress, vest dress, sling dress
outwear	short sleeve outwear, long sleeve outwear

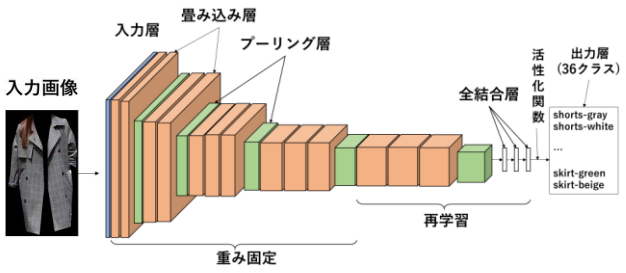


図 1　手法 1 の VGG16 の Fine tuning(bottoms の場合)

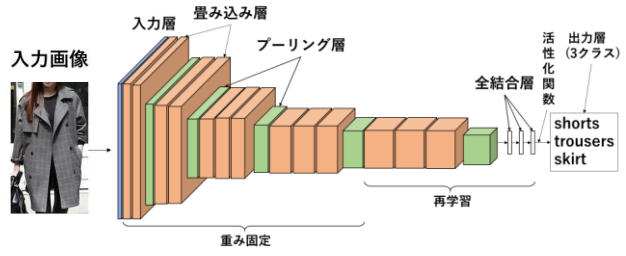


図 2　手法 2 の VGG16 の Fine tuning(bottoms の場合)

4.実験

VGG16 に対して訓練画像を用いて二つの手法で fine tuning を適用した学習後，検証用画像での分類の正解率を算出した結果を表 2 に示す．

表 2 学習後の検証用画像分類の正解率

	tops	bottoms	dress	outwear
手法1	0.659	0.678	0.550	0.704
手法2	0.912	0.938	0.780	0.946

実験の結果，手法 1 が手法 2 より分類の正解率が低いことがわかる．これは，手法 2 では小カテゴリに分類するだけであるのに対し，手法 1 では小カテゴリからさらに色も分類する必要があることが要因とみられる．しかし，色分類の精度が低いことも大きな原因の一つであると考えられる．

次に，提案手法の手法 1 と手法 2 について類似度計算を行い，類似度が高いと表示された上位三つのアイテムの結果例を図 2 に示す．

図 2. 類似度計算を行った結果例

類似度計算を行った複数の結果より判断すると，手法 1 の方が色の類似性は高いが，手法 2 はアイテムの柄などは類似していることが多かった．しかし，被写体のポージングなどが類似度に与える影響が大きく，アイテムが似ていなくても撮影条件次第で類似度が高いと判定されることもあった．

また，手法 1 と手法 2 について推薦の精度を評価するためにアンケートを 12 人に実施した．「対象アイテムと類似していると思うか」という問いに対して 5 段階（5: とても思う，4: 少し思う，3: どちらとも言えない，2: あまり思わない，1: 全く思わない）で評価した．表 3 に評価段階の平均値を示す．

表 3 評価段階の平均値

手法1	手法2
3.39	2.98

これより，手法 1 の方が手法 2 より評価段階の平均値が 0.41 ポイント高いことがわかる．

5. まとめ

本研究では，色と形を考慮した画像特徴量を用いた類似度に基づくファッションアイテムを推薦するシステムを提案した．学習には VGG16 を用い，色と形を考慮した場合と考慮しない場合の 2 通りの手法で学習した．その結果，アンケート評価により，色と形を考慮した学習の方が類似アイテムの推薦システムとして精度が高いことが分かった．

また，さらに推薦精度を上げるには，撮影条件をそろえたファッションアイテムのデータセットの使用及び，色分類手法の再検討が必要である．

参考文献

[1] S. K. Gorripati and A. Angadi, “Visual Based Fashion Clothes Recommendation with Convolutional Neural Networks,” International Journal of Information System and Management Science, vol.1, No.1, pp.1-5, Apr. 2018.

[2] Y. Ge, R. Zhang, L. Wu, X. Wang, X. Tang, and P. Luo, “DeepFashion2: A Versatile Benchmark for Detection, Pose Estimation, Segmentation and Re-Identification of Clothing Images,” arXiv:1901.07973, 2019.

[3] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recogniton,” arXiv: 1409. 1556, 2015.

2019 年度　卒業論文

画像特徴量を用いたファッションアイテム推薦システムの検討

A Study on Fashion Item Recommendation System Using Image Features

指導教員 渡辺 裕 教授

早稲田大学 基幹理工学部
情報通信学科
1W162020-7
安藤　由莉

内容

第1章　序論

1.1 研究背景

近年，ファッション電子商取引業界の市場規模は急速に拡大している．2019 年に経済産業省が発表した 2018 年度の市場規模は 1 兆 7728 億円となっており，対前年度比で 7.7 ポイント上昇した [1]．消費者向け電子商取引における物販系分野の市場規模では 5 カテゴリが 1 兆円を超えているが，その中で衣類が属するカテゴリの市場規模が最も大きい．

このように，衣服をインターネット上で購入する機会は増えた．しかし実店舗で好みの衣服を探すのに比べ，インターネット上で欲しいアイテムを見つけることは，アイテムの数が膨大であることなどから考えて困難である．こうした背景から，これまで一人ひとりの趣味嗜好に合わせたファッションアイテムの推薦に関する研究が多く行われてきた．推薦システムの種類としては，様々な手法が提案されている．あるユーザーの嗜好と類似したユーザーを探し，その嗜好に基づいておすすめを行う方法である協調フィルタリングや，各アイテムの特性を入力として使い，アイテム間の類似性を計算しておすすめする方法であるコンテンツベースフィルタリングが有名である．またその両方を組み合わせたハイブリッドフィルタリングも一般的である．

1.2 研究目的

ユーザーの過去のデータなどを推薦に用いるとどうしても莫大なデータが必要になってしまう．そこで，商品画像のみから推薦するアイテムを選択するコンテンツベースフィルタリングの手法をとる．また，ファッションの重要な要素である色と形という点に注目することで，よりファッションアイテムの推薦として適切なものに近づくと考えられる．本論文では，好みのファッションアイテムを選択するとそのアイテムに類似したアイテムを推薦するシステムを提案する．

1.3 関連研究

視覚的特徴を考慮したファッションアイテムの推薦システムに関する先行研究に，Gorripati らの手法がある．Gorripati らは，CNN モデルの全結合層の特徴量を取り出し，類似度判定を行い推薦するアイテムを提示している [2]．しかし，この手法では色や形を考慮した分類，推薦がされているとは言えない．

1.4 本論文の構成

本論文の構成を以下に示す．

第 1 章は本章であり，本研究の背景，目的，関連研究について述べる．

第 2 章では，本研究で用いる用語についての説明を述べる．

第 3 章では，本研究で提案する手法について述べる．

第 4 章では，第 3 章で述べた手法に対する評価実験の結果について述べる．

第5章では本研究の結論と今後の課題を述べる.

第2章　関連用語

2.1 まえがき

本章では，本研究に関連する用語について説明する．ニューラルネットワークと，畳み込みニューラルネットワークである Convolutional Neural Network (CNN)，CNN に用いた fine tuning，fine tuning に用いたニューラルネットワークモデルである VGG16 について述べる．

2.2 ニューラルネットワーク

ニューラルネットワークとは，ノードを何層にもわたってつなぎ合わせて構築することでできるネットワークである [3] [4]．ノードとは，複数の信号を入力として受け取り，ひとつの信号を出力する接続点である．図 2.1 にニューラルネットワークの構造例を示す．ニューラルネットワークには複数の入力と複数の出力があり，入力層，中間層，出力層というように層が並んでいる．各層は複数のノードを持ち，2 層間のノードはエッジで結ばれている．各ノードは値を持ち，各エッジは重みを持つ．あるノードの値は一つ前の層のそのノードに接続されているノードの値，それらを接続するエッジの重み，その層が持つ活性化関数から計算される．ディープラーニングにおいては，ニューラルネットワークの重みパラメータが最適化されることで，学習が行われる．

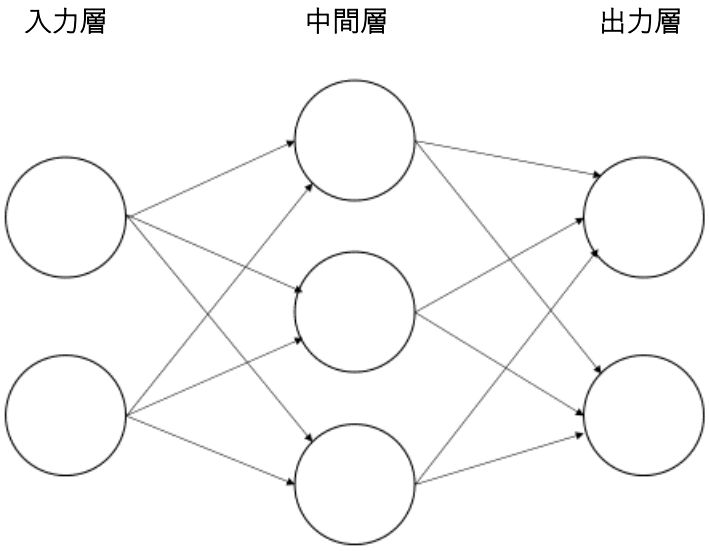


図 2.1　ニューラルネットワークの構造例

2.3 CNN

CNN とは，多くの畳み込み層を持つニューラルネットワークで，特に画像認識の分野で優れている [3] [4] [5] [6]．CNN は入力層，畳み込み層，プーリング層，全結合層，出力層から構成される．CNN の例を図 2.2 に示す．

畳み込み層で行う処理は「畳み込み演算」であり，入力データに対してフィルタを適用する．入力データに対してフィルタのウィンドウを一定の間隔でスライドさせながら適用し，その結果を出力の対応する場所へ格納していく．

プーリング層では空間サイズを小さくする演算を行う．例えば，2×2 の領域を一つの要素に集約するような処理を行い，情報を圧縮することで，入力データの微小な位置変化に対して頑健となる．

全結合層では隣接する層のすべてのノード間で結合があるのに対し，畳み込み層ではノード同士の結合をうまく制限している．なおかつ重み共有という手法をとることで画像の畳み込みに相当するような処理をニューラルネットワークの枠組みの中で表現することに成功している．

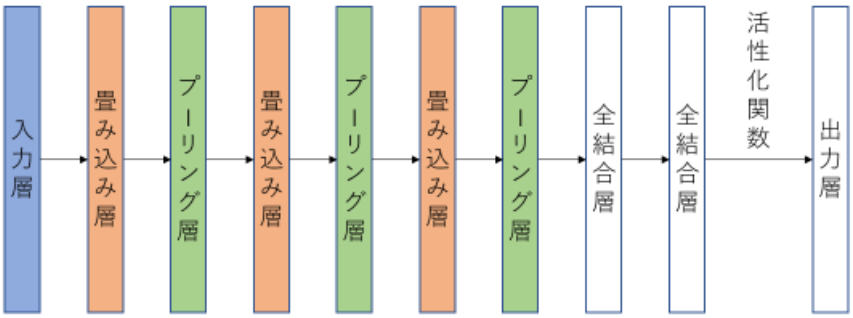


図 2.2　CNN の構造例

2.4 VGG16

VGG16 とは，ImageNet データベースの 100 万枚を超える画像で学習済みの畳み込みニューラルネットワークである [7] [8]．このネットワークは，深さが 16 層あり，画像を 1000 個のオブジェクトカテゴリに分類できる．結果として，このネットワークは広範囲のイメージに対する豊富な特徴表現を学習している．図 2.3 に VGG16 モデルの構造を示す．層の大きさは出力値の次元サイズの大きさを表す．

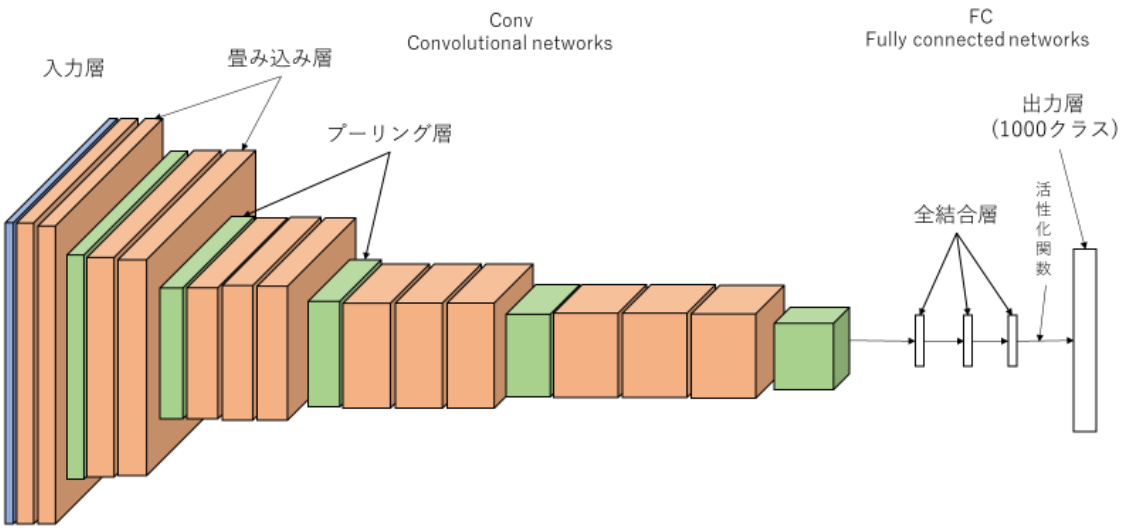


図 2.3　VGG16 モデルの構造

2.5 fine tuning

fine tuning とは，すでに学習済みのモデルを転用して，新たなモデルを生成する手法である [9]．つまり，他の画像データを使って学習されたモデルを使うことによって，新たに作るモデルは少ないデータや学習量でモデルを生成することが可能となる．図 2.4 に fine tuning の例を示す．

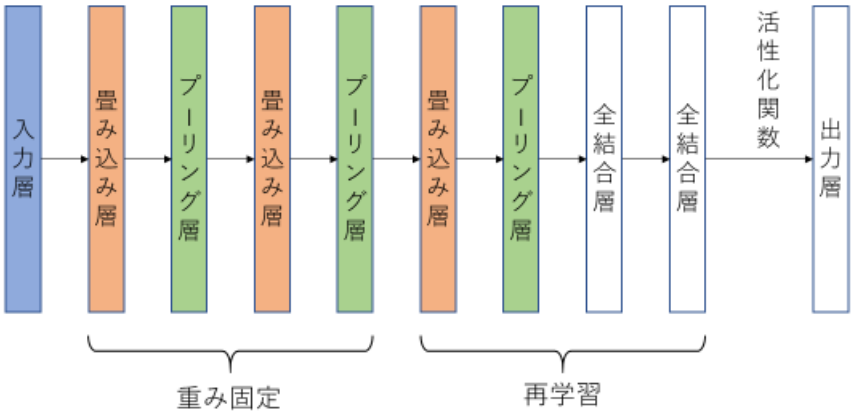


図 2.4　fine tuning の例

2.6 コサイン類似度

コサイン類似度とは，ベクトルの内積を用いて類似度を計算する方法である［10]．-1 から 1 までの値を取り，数値が大きくなるほど類似度が高い．

個体iの変量ベクトルを$\boldsymbol{x}_i$，個体jの変量ベクトルを$\boldsymbol{x}_j$としたとき，個体iと個体jの類似度S_{ij}は次の式(2.1)のように計算できる．

$$S_{ij} = \frac{\boldsymbol{x}_i \cdot \boldsymbol{x}_j}{|\boldsymbol{x}_i||\boldsymbol{x}_j|} = \frac{\sum_{k=1}^{m} x_{ki} x_{kj}}{\sqrt{\sum_{k=1}^{m} x_{ki}^2 \sum_{k=1}^{m} x_{kj}^2}} \tag{2.1}$$

2.7 むすび

本章では，本研究で用いるディープラーニングの技術であるニューラルネットワーク，CNN，VGG16，fine tuning 及びコサイン類似度などの用語について説明した．

第3章　提案手法

3.1 まえがき

本章では，本研究で提案する手法について述べる．

3.2 データの前処理

本節では，本研究で使用するデータセットと，それに対する前処理について述べる．

3.2.1　使用するデータセット

本研究では，ファッションのデータセットである DeepFashion2 を用いた [11]．DeepFashion2 は電子商取引サイトとそれを利用する消費者の両方を情報源としており，以下の情報が含まれている．13 の衣服カテゴリのイメージ画像と，バウンディングボックスやセグメンテーション領域の座標，スケール（Scale，あるアイテムの写真の中で占める割合），オクルージョン（Occlusion，あるアイテムが何らかの物体によってどれだけさえぎられているか）や拡大（Zoom-in）の度合い，視点（View point）などがある．表 3.1 にデータセット画像の統計を，図 3.1 に 13 の衣服カテゴリとそれぞれのデータの数を示す．

このデータセットには全部で 49 万枚の画像がある．このうち情報源が EC サイトの画像でかつ Occlusion が slight または medium，Zoom-in が no または medium である約 18 万枚の画像のみを学習に用いた．

表 3.1 データセット画像の統計

Scale	small : 26%	moderate : 50%	large : 24%	
Occlision	slight : 47%	medium : 47%	heavy : 6%	
Zoom-in	no : 67%	medium : 21%	large : 12%	
Viewpoint	no wear : 7%	frontal : 78%	side : 8%	back : 7%

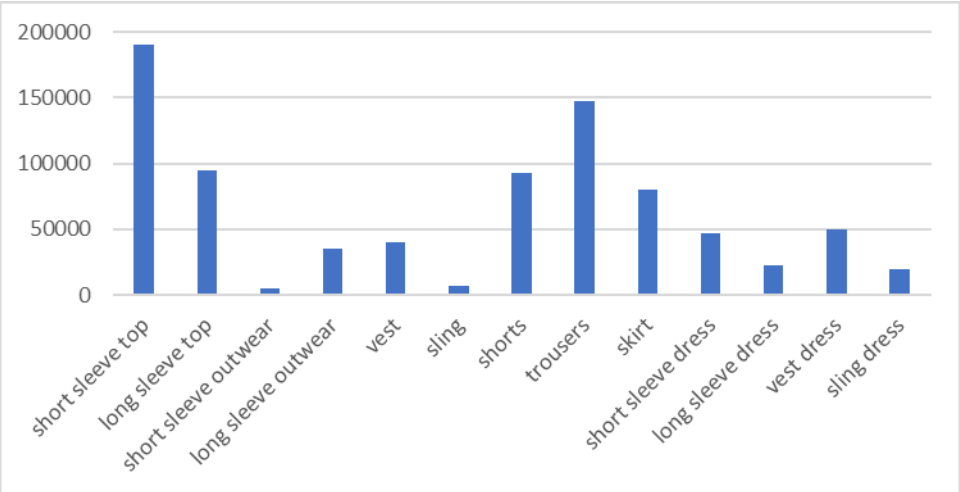


図 3.1　13 の衣服カテゴリとそれぞれのデータの数

3.2.2　データセットの衣服領域セグメンテーション

DeepFashion2 データセットに含まれるセグメンテーション領域の座標から，対象のアイテムの領域のみ切り出し，それ以外を黒く塗りつぶす．図 3.2 に元画像とセグメンテーション後の画像の例を示す．

図 3.2 元画像とセグメンテーション後の画像の例（“Deep fashion2” [11]）

3.2.3　データセットの色分類

データセットの画像からアイテムごとにセグメンテーションの領域内ピクセルに含まれる最も多い色に基づいて色分けをする．このとき，分ける色はファッションアイテムの分類として適切であるように，大手ファッション EC サイト ZOZOTOWN [12]の色分けを参考にした．グレー，ホワイト，ブラック，レッド，ブラウン，ピンク，オレンジ，イエロー，パープル，ブルー，グリーン，ベージュの 12 色とした．

色分けの手法については，本研究で扱うカラー画像は赤成分(R)，緑成分(G)，青成分(B)それぞれ各 8 ビットあり，色数は 256×256×256=16777216 通りあり、冗長であるので減色作業を行う．減色は RGB の各成分の値を 0~31 を 16 に，32~95 を 64 に，96~159 を 128 に，160~223 を 192 に，224~255 を 240 として代表値に置き換える．これにより、色数は 5×5×5=125 通りとなる．125 通りの色に番号を付けるため，RGB の値を置き換えた代表値 16，64，128，192，240 をそれぞれ 0，1，2，3，4 とさらに置き換えた．0~4 の番号で

置き換えた赤成分，緑成分，青成分の変数をそれぞれ r，g，b として以下の式(3.1)の計算を行う．図 3.3 に RGB の値の変換値を示す．

$$color = 25r + 5g + b \qquad (3.1)$$

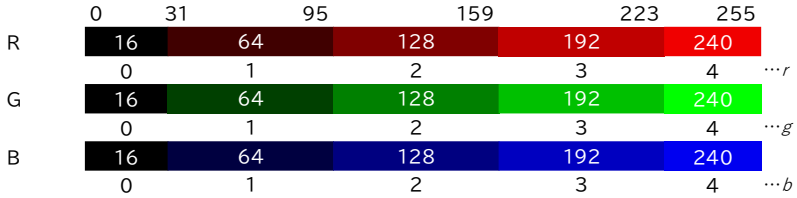


図 3.3 RGB の値の変換値

さらにこの 125 通りの色を，自分で目視して上記の 12 色に分類した．表 3.2 に分類した結果を示す．

表 3.2 色分類した結果

color		color		color		color		color	
0	ブラック	25	ブラウン	50	ブラウン	75	レッド	100	レッド
1	ブルー	26	パープル	51	レッド	76	レッド	101	レッド
2	ブルー	27	パープル	52	パープル	77	ピンク	102	ピンク
3	ブルー	28	ブルー	53	パープル	78	パープル	103	ピンク
4	ブルー	29	ブルー	54	パープル	79	パープル	104	パープル
5	グリーン	30	グリーン	55	ブラウン	80	オレンジ	105	オレンジ
6	グリーン	31	グレー	56	ブラウン	81	レッド	106	オレンジ
7	ブルー	32	ブルー	57	パープル	82	ピンク	107	ピンク
8	ブルー	33	ブルー	58	パープル	83	パープル	108	ピンク
9	ブルー	34	ブルー	59	パープル	84	パープル	109	パープル
10	グリーン	35	グリーン	60	グリーン	85	イエロー	110	オレンジ
11	グリーン	36	グリーン	61	グリーン	86	ベージュ	111	オレンジ
12	ブルー	37	ブルー	62	グレー	87	ピンク	112	ピンク
13	ブルー	38	ブルー	63	ブルー	88	パープル	113	ピンク
14	ブルー	39	ブルー	64	ブルー	89	パープル	114	ピンク
15	グリーン	40	グリーン	65	グリーン	90	イエロー	115	イエロー
16	グリーン	41	グリーン	66	グリーン	91	イエロー	116	イエロー
17	グリーン	42	グリーン	67	グリーン	92	ベージュ	117	ベージュ
18	ブルー	43	ブルー	68	ブルー	93	グレー	118	ベージュ
19	ブルー	44	ブルー	69	ブルー	94	ブルー	119	ピンク
20	グリーン	45	グリーン	70	グリーン	95	イエロー	120	イエロー
21	グリーン	46	グリーン	71	グリーン	96	イエロー	121	イエロー
22	グリーン	47	グリーン	72	グリーン	97	イエロー	122	イエロー
23	グリーン	48	グリーン	73	グリーン	98	イエロー	123	ベージュ
24	ブルー	49	ブルー	74	ブルー	99	ブルー	124	ホワイト

色分けをする対象アイテム画像のセグメンテーション領域内のピクセルの RGB 値を上記の手法で 12 通りに分類する．最も多かった色をそのアイテムの色と決定づける．

3.3 VGG16 モデルでの学習

ファッションアイテムの上記の 13 カテゴリ（小カテゴリとする）を以下の表 3.3 のように 4 グループに分け，これを大カテゴリとする．大カテゴリごとに VGG16 で学習する．

表 3.3 カテゴリ分け

大カテゴリ	小カテゴリ
tops	short sleeve top, long sleeve top, vest, sling
bottoms	shorts, trousers, skirt
dress	short sleeve dress, long sleeve dress, vest dress, sling dress
outwear	short sleeve outwear, long sleeve outwear

3.3.1 手法 1：色と形を考慮して学習した場合

アイテムを小カテゴリで分類し，さらにその中で，3.2.3 節の手法で色分類したクラスを出力層として，VGG16 モデルを用いて fine tuning を行う．入力画像はセグメンテーション後の画像を用いる．図 3.4 に大カテゴリが bottoms の場合の手法 1 の fine tuning を行った VGG16 モデルの構造を示す．

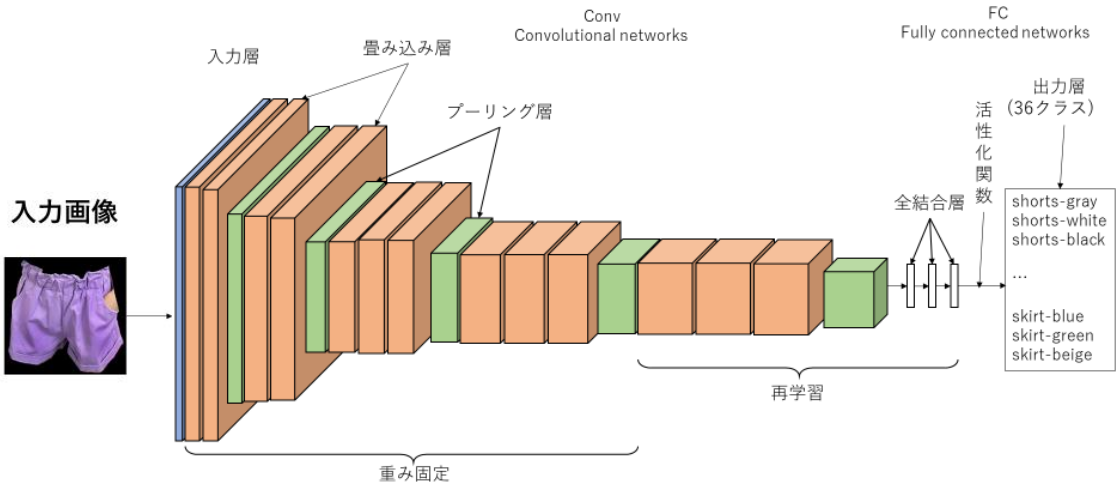


図 3.4 手法 1 の VGG16 の Fine tuning(bottoms の場合，“Deep fashion2” [11])

このように，三つの畳み込み層，一つのプーリング層，全結合層を再学習する．図 3.5 に大カテゴリが bottoms の場合の手法 1 のモデル結果を示す．

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 224, 224, 3)       0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, 224, 224, 64)      1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, 224, 224, 64)      36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, 112, 112, 64)      0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, 112, 112, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, 112, 112, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, 56, 56, 128)       0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, 56, 56, 256)       295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, 28, 28, 256)       0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, 28, 28, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, 14, 14, 512)       0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, 7, 7, 512)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 25088)             0
_________________________________________________________________
dense_1 (Dense)              (None, 4096)              102764544
_________________________________________________________________
dropout_1 (Dropout)          (None, 4096)              0
_________________________________________________________________
dense_2 (Dense)              (None, 36)                147492
=================================================================
Total params: 117,626,724
Trainable params: 117,626,724
Non-trainable params: 0
```

図 3.5 手法 1 のモデル結果（bottoms の場合）

学習終了後，もう一度すべての訓練画像を fine tuning した VGG16 モデルに通す．出力層の直前の全結合層である図 3.5 の dense_1 層の 4096 次元の値を取り出し，大カテゴリごとのファイルに保存する．

3.3.2　手法 2：色と形を考慮せずに学習した場合

3.3.1 節と同様にアイテムは小カテゴリで分類するが，色分類はせずに VGG16 モデルを用いて fine tuning を行う．入力画像はセグメンテーション前の画像で，アイテムをバウンディングボックスで切り取ったものを用いる．図 3.6 に大カテゴリが bottoms で色と形を考慮しない場合の fine tuning を行った VGG16 モデルの構造を示す．

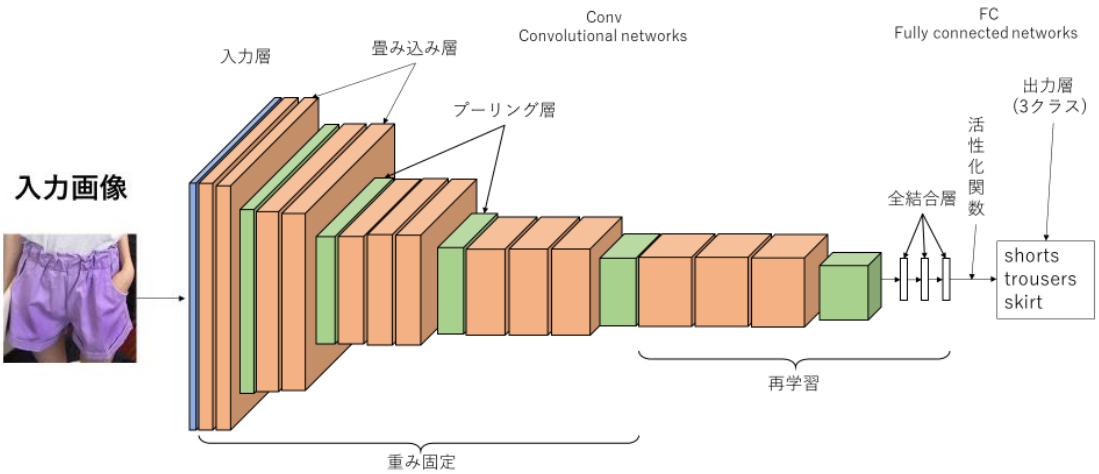
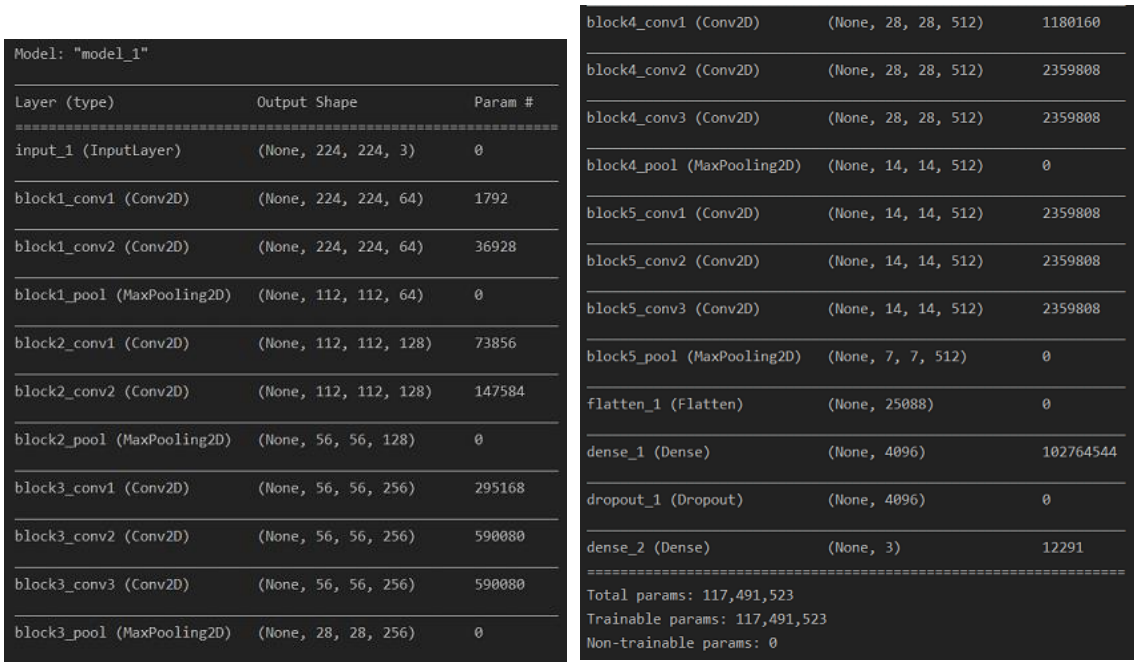


図 3.6　手法 2 の VGG16 の fine tuning（bottoms の場合，“Deep fashion2” [11]）

最終的な出力層は bottoms の小カテゴリ 3 クラスになる．図 3.7 に大カテゴリが bottoms で色と形を考慮しないときのモデル結果を示す．



Model: "model_1"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten_1 (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 4096)	102764544
dropout_1 (Dropout)	(None, 4096)	0
dense_2 (Dense)	(None, 3)	12291

Total params: 117,491,523
Trainable params: 117,491,523
Non-trainable params: 0

図 3.7 手法 2 のモデル結果（bottoms の場合）

3.3.1 節と同様に，学習終了後，もう一度すべての訓練画像を fine tuning した VGG16 モデルに通す．出力層の直前の全結合層である図 3.7 の dense_1 層の 4096 次元の値を取り出し，大カテゴリごとのファイルに保存する．

3.4 類似度計算

ある一つのアイテムを選んだ時，そのアイテムと同じ大カテゴリのすべてのアイテムについて 4096 次元ベクトルのコサイン類似度計算を行い，最も類似度が高いものから順に表示する．

3.5 むすび

本章では，本研究で提案するファッションアイテム推薦システムの手法について述べた．

第4章　実験

4.1 まえがき

本章では，色と形を考慮して VGG16 モデルに fine tuning を適用し、学習した場合（手法 1）と考慮せずに学習した場合（手法 2）の，学習後の分類の正解率と、類似度計算をした結果，提案手法に対する評価実験とその結果，及びそれらに対する考察を述べる．

4.2 VGG16 による学習結果

VGG16 に対して訓練画像を用いて二つの手法で fine tuning を適用した学習後，検証用画像での分類の正解率を算出した結果を表 4.1 に示す．実験の結果、手法 1 の方が手法 2 よりも分類の正解率は低くなっている．

表 4.1　学習後の検証用画像分類の正解率

	tops	bottoms	dress	outwear
手法1	0.659	0.678	0.550	0.704
手法2	0.912	0.938	0.780	0.946

4.3 類似度の計算結果

提案手法の類似度計算を行った実験の結果例を図 4.1 に示す．

図 4.1　類似度計算の結果例（“Deep fashion2” [11]）

4.4 評価実験

4.4.1 評価実験の内容

手法 1 と手法 2 について，推薦の精度を評価するためにアンケートを実施した．アンケートは 3 種類，それぞれ 4 人に，全部で 12 人に実施した．一つのアンケートにつきランダムに 26 個のアイテム（tops8 個，bottoms6 個，dress8 個，outwear4 個）を選び，類似度計算する. 手法 1 と手法 2 について上位三つの類似度が高いと表示されたアイテムを合わせ，順序を変えて 6 枚の写真を並べる．対象アイテムと 6 枚の写真を見比べ，「対象アイテムと類似していると思うか」という問いに対して 5 段階（5: とても思う，4: 少し思う，3: どちらとも言えない，2: あまり思わない，1: 全く思わない）で評価した．

4.4.2 評価実験の結果

手法 1 と手法 2 について，5 段階評価の平均値を表 4.2 に示す．その結果、手法 1 の方が手法 2 よりも 0.41 ポイント高いことがわかる．

手法1	手法2
3.39	2.98

表 4.2　評価段階の平均値

すべてのアイテムについて被験者の回答をまとめ，手法 1 と手法 2 の評価段階の平均値を計算する．平均値を比較した結果を表 4.3 に示す．これより、手法 1 の方が評価段階の平均値が高いアイテム数が多いことがわかる．

手法1の方が評価段階の平均値が高かったアイテム数	手法2の方が評価段階の平均値が高かったアイテム数	平均値が同じだったアイテム数
48	27	3

表 4.3　アイテムごとの平均値の比較結果

4.5 考察

4.5.1 VGG16 による学習結果の考察

4.2 節より VGG16 による学習結果について考察する．手法 1 の方が手法 2 より分類の正解率が低いのは，手法 2 では小カテゴリに分類するだけであるのに対し，手法 1 では小カテゴリからさらに色も分類する必要があることが要因とみられる．しかし，色分類の精度が低いことも大きな原因の一つであると考えられる．実際に色分類されたファイルを観察す

ると，ある色のファイルに，違う色のファイルに存在するべき画像があることが多々あった．したがって，色分類手法の再検討が必要である．

4.5.2 類似度の計算結果の考察

4.3 節より類似度の計算結果について考察する．図 4.1 は手法 1 と手法 2 について類似度計算をした結果の一例であり，他の複数の結果より判断すると，手法 1 と手法 2 では手法 1 の方が色の類似性が高いことがわかった．一方，手法 2 はアイテムの柄やプリントなどが類似していることが多かった．しかし，被写体の体の向きやポージングなどが類似度に与える影響が大きく，アイテム自体が似ていなくても撮影条件次第で類似度が高いと判定されるケースもあった．

4.5.3 評価実験の結果の考察

4.4 節より評価実験の結果について考察する．表 4.2，表 4.3 から， 色と形を考慮して学習した手法 1 の方が，色と形を考慮せずに学習した手法 2 より類似アイテムの推薦システムとして精度が高いことが分かる．このことから，ファッションアイテムの推薦において色と形を考慮することが重要であると考えられる．

4.6 むすび

本章では，3 章の提案手法における VGG16 の学習と類似度の計算の結果，提案手法の評価実験及びその結果について述べた．また，それぞれの実験結果に対する考察を述べた．

第5章　結論と今後の課題

5.1 結論

本研究では，色と形を考慮した画像特徴量を用いた類似度に基づくファッションアイテムを推薦するシステムを提案した．学習には VGG16 を用い，色と形を考慮した場合と考慮しない場合の 2 通りの手法で学習した．その結果，アンケート評価により，色と形を考慮した学習の方が類似アイテムの推薦システムとして精度が高いことが分かった．

5.2 今後の課題

VGG16 での学習結果において，検証用画像の分類精度が低い原因は，色分類の手法が理想的ではないためであると考えられる．また，類似度計算の結果から，ファッションアイテムを身に着けている人のポージングや，撮影条件などに類似度は大きく影響を受けていることが分かった．これらの問題を改善し，さらに推薦の精度を上げるためには，撮影条件をそろえたファッションアイテムのデータセットの使用及び，色分類手法の再検討が必要である．

謝辞

本研究の実験環境を整えてくださり，研究の方向性について丁寧かつ熱心なご指導を頂いた渡辺裕教授と笠井裕之教授に心から感謝いたします.

また，日頃から御意見やアドバイスをくださった研究室の皆様に御礼申し上げます.

最後に，私をここまで育ててくださった家族に感謝いたします.

参考文献

[1] 経済産業省，“平成 30 年度 我が国におけるデータ駆動型社会に係る基盤設備（電子商取引に関する市場調査について），” 2019 年 5 月，https://www.meti.go.jp/press/2019/05/20190516002/20190516002-1.pdf. 2020 年 1 月閲覧.

[2] S. K. Gorripati and A. Angadi, “Visual Based Fashion Clothes Recommendation with Convolutional Neural Networks,” International Journal of Information System and Management Science, vol.1, No.1, pp.1-5, Apr. 2018.

[3] 斎藤康毅，ゼロから作る Deep Learning－Python で学ぶディープラーニングの理論と 実装，pp.39-44，205-220，オライリー・ジャパン，2016.

[4] 武井宏将，初めてのディープラーニング－オープンソース"Caffe"による演習付き，リックテレコム, pp.6-9, 2016.

[5] “畳み込みニューラルネットワーク（CNN），” MathWorks, https://jp.mathworks.com/discovery/convolutional-neural-network.html. 2020 年 1 月閲覧.

[6] システムインテグレータ，“畳み込みニューラルネットワーク_CNN(Vol.16)，” AISIA，2018 年 5 月 11 日. https://products.sint.co.jp/aisia/blog/vol1-16. 2020 年 1 月閲覧.

[7] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recogniton,” arXiv: 1409. 1556, 2015.

[8] “vgg16,” MathWorks, https://jp.mathworks.com/help/deeplearning/ref/vgg16.html;jsessionid=755bf9524bcaa5696a605a323629. 2020 年 1 月閲覧.

[9] “少ない画像から画像分類を学習させる方法（keras で転移学習：fine tuning），” SPJ, 2020 年 1 月 26 日. https://spjai.com/keras-fine-tuning/. 2019 年 1 月閲覧.

[10] “統計用語集 コサイン類似度,“ BellCurve 統計 WEB, https://bellcurve.jp/statistics/glossary/5488.html. 2020 年 1 月閲覧.

[11] Y. Ge, R. Zhang, L. Wu, X. Wang, X. Tang, and P. Luo, “DeepFashion2: A Versatile Benchmark for Detection, Pose Estimation, Segmentation and Re-Identification of Clothing Images,” arXiv:1901.07973, 2019.

[12] “ZOZOTOWN,” ZOZO, Inc., https://zozo.jp/. 2020 年 1 月閲覧.

図一覧

表一覧