社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

[サーベイ論文] Deep Learningを用いた歩行者検出の研究動向

福井　　宏†　山下　隆義†　山内　悠嗣†　藤吉　弘亘†

† 中部大学　〒 487–8501 愛知県春日井市松本町 1200

E-mail: †fhiro@vision.cs.chubu.ac.jp, ††{yamashita,hf}@cs.chubu.ac.jp, †††yuu@isc.cs.chubu.ac.jp

あらまし　2012 年の Deep Learning 到来以降，歩行者検出の分野においても勾配ベース手法から Convolutional Neural Network (CNN) を利用した手法に大きく移り変わっている．CNN をベースにした手法は，学習過程において歩行者検出に有効な特徴抽出が可能であり，歩行者検出のベンチマークで高い性能を実現している．そこで，本稿では CNN をベースにした歩行者検出法についてサーベイし，1) 2 段階の検出構造による歩行者検出と，2) Region proposal ベースによる歩行者検出の観点において各手法について述べる．

キーワード　Deep Learning，歩行者検出，サーベイ

1. はじめに

歩行者検出は，カメラで得られたフレームから歩行者の位置と大きさを推定する技術であり，歩行者の向きや姿勢，服装などの見えの変化に頑健な特徴量を設計する必要がある．山内らは，2012 年までの歩行者検出法を調査することで，歩行者検出が困難になる要因を明確にし，その要因を解決するアプローチをまとめている [1]．2012 年までの歩行者検出は，Histogram of Oriented Gradient (HOG) 特徴量 [2] が用いられており，歩行者の勾配方向ヒストグラムを特徴量として扱うことで，歩行者の向きや姿勢，服装などの見えの変化に頑健な検出を実現した．Dalal らが HOG 特徴量を提案後，HOG 特徴量をベースにした手法が数多く提案されている [3] [4] [5]．2009 年には，色情報や勾配情報等を用いたチャンネル特徴量と Boosted tree を組み合わせた Integral Channeled Feature (ICF) [6] が提案されている．複数の特徴抽出手法を組み合わせたチャンネル特徴量を用いることで検出性能が大幅に向上し，ICF をベースにした歩行者検出法が提案された [7] [8] [9]．2012 年に開催された，ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [10] の一般物体認識コンテストにおいて Deep Convolutional Neural Network (CNN) を用いた方法がトップになって以降 [11]，CNN を歩行者検出に応用した手法が数多く提案されている [12]〜[25].

CNN による歩行者検出は，1) 2 段階の検出構造による方法と 2) Region proposal ベースによる方法の 2 つに大別できる．CNN による 2 段階の検出構造は，はじめに別の識別器で歩行者候補領域を検出し，検出した歩行者候補領域を CNN に入力することで識別する．このような 2 段階の検出構造を用いる理由は 2 つある．1 つ目は，CNN による計算コストを削減するためである．歩行者検出は一般的にラスタスキャンによりスケールの異なる複数の検出枠をずらしながら検出するため，1 枚のフレームを処理するのに数万単位の識別処理が必要になる．CNN は非常に計算コストが高く，1 枚のフレームに対して数万単位の識別処理を CNN で行うのは非現実的である．2 つ目は，誤検出を削減するためである．背景サンプルは外観の変化が歩行者より大きいため，CNN の学習時に様々な背景画像を学習させる必要がある．しかし，様々な背景を網羅した学習サンプルを用意し，CNN を学習することは非現実的である．そのため，2 段階の検出処理により CNN の検出対象を限定し，高精度化を実現している．2015 年には，Fast R-CNN [26]，Faster R-CNN [27]，Single Shot MultiBox Detector [28] 等の Region proposal による歩行者検出法が提案されている．これらの手法は，1 つの CNN のみ用いて検出処理を行うため，2 段階の検出構造にする必要がなく，高速に歩行者を検出できる．

このように，Deep Learning の発展に伴い，CNN をベースにした歩行者検出法が数多く提案されている．本稿では山内らのサーベイ以降に提案されている Deep Learning をベースとした歩行者検出法についてサーベイし，Deep Learning が歩行者検出にどのように応用されているかを理解する．本稿の構成は，2 章で歩行者検出に用いられる特徴量や識別器の移り変わりについて述べる．そして，歩行者検出の性能を評価する際に用いられるデータセットについて，3 章で述べる．2 段階の検出構造による歩行者検出に用いられる手法ついては 4 章，Region proposal ベースの歩行者検出法については 5 章で述べる．

2. 歩行者検出法の歴史

歩行者検出の初期の研究では，切り出した歩行者画像と背景画像の認識問題として扱われ，Wavelet 特徴量と SVM の組み合わせにより歩行者認識を行っていた [29]．そして，1 枚のフレームから歩行者の位置と大きさを推定する検出問題へ遷移し，2004 年に提案された Viola らの顔検出の研究が歩行者検出に応用された [30]．しかし，濃淡のみ用いる Haar-like 特徴量は，顔検出の分野では高い性能を実現しているが，歩行者検出では十分な性能を得られなかった．そのため，2005 年に Dalal らは歩行者の

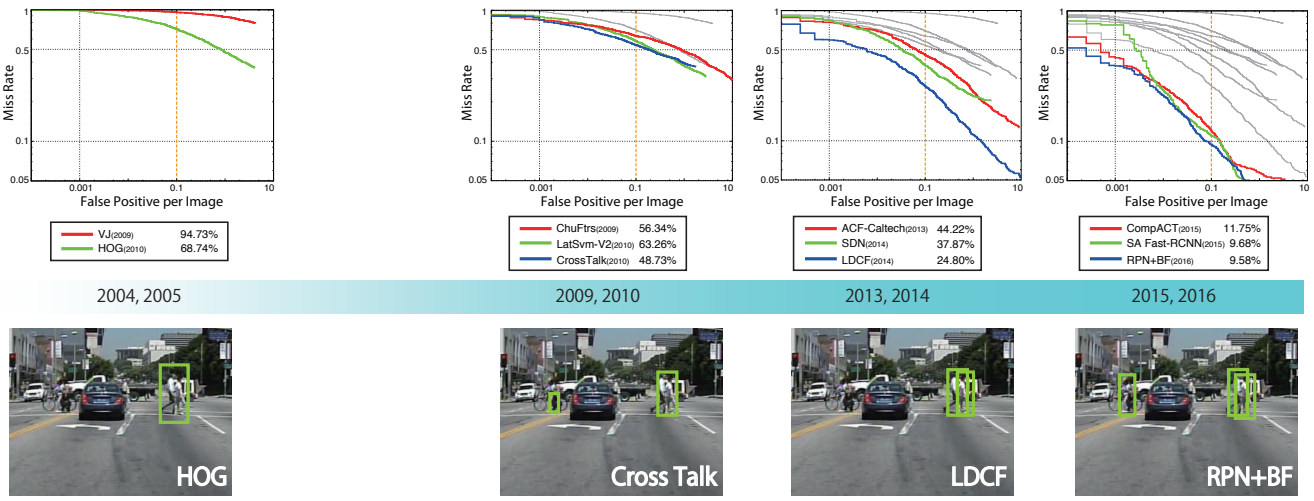


図 1　歩行者検出性能の遷移

勾配に着目する Histogram of Oriented Gradient (HOG) 特徴量を提案し，Support Vector Machine (SVM) と組み合わせることで，歩行者検出の性能を大幅に向上させた [2]．Deformable Part Model (DPM) [3] は，解像度の異なる HOG 特徴量から，歩行者の全身と身体のパーツを捉えて検出するため，姿勢の変化に頑健な歩行者検出を実現した．

2009 年には，HOG 特徴量の勾配方向ヒストグラムだけでなく，色情報等を特徴量に加えるチャンネル特徴量を用いた ICF が提案された [6]．チャンネル特徴量は，勾配情報や色情報などを複数の画像のチャンネルのように扱う特徴量である．また，識別器に Boosted tree を用いることで，膨大な特徴量群から歩行者検出に有効な特徴量を選択することができる．ICF にチャンネル特徴量の Aggregate (集約) 処理を加えた Aggregate Channel Feature (ACF) [8] は，各チャンネル特徴量に対して特徴ピラミッドを導入することで，歩行者のスケール変化に頑健な歩行者検出を実現した．

2012 年に一般物体認識のコンペティションである ILSVRC [10] で，Krizhevsky らの CNN をベースにした手法 [11] がトップとなり，歩行者検出においても CNN をベースとした手法が数多く提案されている [12]〜[25]．CNN による歩行者検出の先駆けとなる Joint Deep Learning [13] は，入力画像を CNN に入力することで特徴マップを獲得し，パーツ領域ごとに後段の Restricted Boltzmann Machine (RBM) [13] へ入力することで歩行者を検出する．その後，Luo らは歩行者検出に有効なパーツ領域を学習により選択する Switchable Deep Network (SDN) [14] を提案した．これらの手法は，2 段階の検出構造で実現されている．2 段階の検出は，はじめに他の識別器で歩行者候補領域を検出し，後段の CNN で検出した歩行者候補領域に対して歩行者か否かを識別する．2 段階の検出構造における初期の研究では，歩行者のパーツを学習させる CNN を構築し，歩行者検出の精度を向上させた．その後，歩行者のパーツ情報ではなく，歩行者と背景の様々な属性を学習することで，歩行者検出性能を向上させた Task Assistance-CNN (TA-CNN) が

図 2　歩行者検出のデータセット例

提案された．2015 年には，複数の識別器をカスケード状に配置した Deep cascade や，特徴抽出においてカスケード構造を導入した Complexity-Aware Cascade Training (CompACT) [21] が提案され，CNN による歩行者検出の計算コスト削減に関する研究も取り組まれ始めた．CNN から特徴マップを特徴量として扱う研究や，CNN による歩行者検出の計算コスト削減に関する研究も取り組まれ始めた．

また，Fast R-CNN [26] や Faster R-CNN [27]，Single Shot Detector (SSD) [28] をベースにした歩行者検出法も提案されている [22] [21] [27] [25]．これら歩行者検出法は，スケールの小さい歩行者に対応するために，CNN の各層でスケールの異なる歩行者を検出したり，アンカーの走査で使用する特徴マップの解像度を変更するなどの工夫を施している．これまでに述べた，2004 年から 2016 年における歩行者検出の性能の遷移を図 1 に示す．図 1 の検出精度は，Caltech Pedestrian Benchmark [31] で比較している．図 1 より，HOG+SVM の検出精度が 68.74% で Fused DNN の検出精度が 8.2% であり，59.16% 検出精度が向上していることがわかる．

3.　歩行者検出のデータセット

歩行者検出法を同じ基準で評価するために，表 1 のような様々

表 1　各データセットの詳細

	# of Ped.	学習のフレーム数	評価のフレーム数	撮影環境	撮影時間	オクルージョン情報	追加情報
Caltech [31]	学習：192,000 評価：155,000	33,171	4,024	走行シーン	3 時間	あり	
ETH [32]	学習：– 評価：12,000	–	1,101	市街地	5 分	あり	ステレオ情報 トラッキング
Daimler [33]	学習：15,600 評価：56,500	Ped.：15,560 Bkg.：6,744	21,790	走行シーン	27 分 (評価のみ)	あり	ステレオ情報 セグメンテーション
KITTI [34]	25,000	80,000		走行シーン	6 時間	あり	ステレオカメラ LIDAR 地図情報 物体の向き トラッキング

な歩行者検出のデータセットが公開されている．Dalal らは，HOG 特徴量と SVM の組み合わせを評価するために，INRIA Person Dataset [2] を構築した．INRIA Person Dataset は，約 2,200 枚程度の小規模なデータセットであり，デジタルカメラで撮影された画像から構築されている．しかし，ICF や CNN など学習サンプルを多く必要とする統計的学習法が用いられるようになり，Caltech Pedestrian Dataset [31] や KITTI Dataset [34] のような大規模なデータセットが用いられるようになった．本章では，歩行者検出の評価で用いられる各データセットと，その特性について述べる．

3.1　Daimler Pedestrian Benchmark Data Sets [33]

Daimler Pedestrian Benchmark Data Sets は Daimler 社が作成した初期の歩行者検出データセットであり，車載カメラで撮影した動画をベースにデータセットを構築されている．Daimler Pedestrian Benchmark Data Sets は，グレースケール画像で構築されており，切り出して正規化された歩行者画像と背景画像からデータセットが構築されている．撮影場所は，市街地や人工物の少ない道路や公園，建物内の駐車場など様々な箇所を撮影している．データセットは Caltech Pedestrian Dataset と同程度の規模であり，学習サンプルと評価サンプルで約 36,000 枚の歩行者が存在している．2010 年以降に Daimler Pedestrian Data Sets は，ステレオカメラから取得した視差マップや，シーンセグメンテーションのデータも公開している．

3.2　INRIA Person Dataset [2]

INRIA Person Dataset は，デジタルカメラで市街地の歩行者を撮影したシーンから構成されている歩行者検出のデータセットである．INRIA Person Dataset は他のデータセットと比べて，人の領域が大きく解像度が高いため，他のデータセットと比べて検出が容易である．INRIA Person Dataset の特徴は，撮影されている人物の対象が歩行者のみでなく，図 2(b) のように運動をしている人物など姿勢の変化が他のデータセットと比べて大きい．

3.3　ETH Dataset [32]

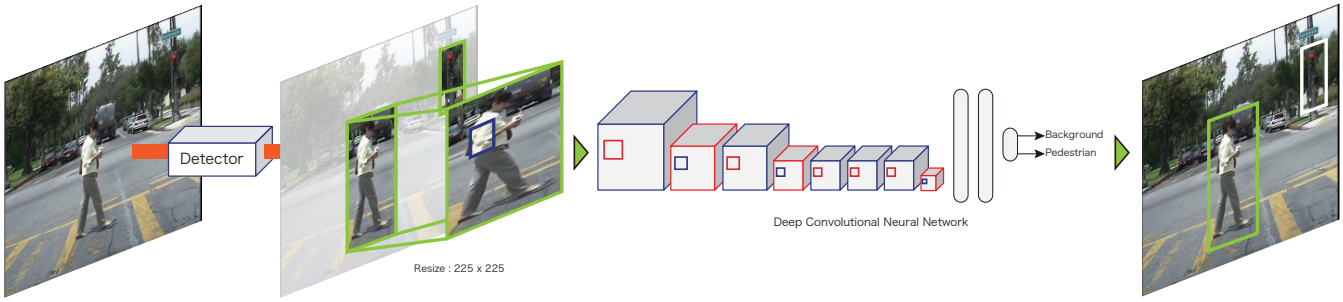
ETH Dataset は，人混みが多い市街地を撮影した動画から構成されている．また，RGB 画像のみでなく，ステレオカメラで撮影した距離画像も公開されており，歩行者の 3 次元位置とトラッキング情報も教師ラベルとして用意されている．ETH Dataset の規模は約 1,000 枚程度であるが，図 2(c) のような群衆のシーンが多いため検出が困難である．また，ETH Dataset には学習用画像が用意されておらず，歩行者の見えが似ている INRIA Person Dataset [2] が学習サンプルとして一般的に用いられる．

3.4　Caltech Pedestrian Dataset [31]

Caltech Pedestrian Dataset は，車載カメラで撮影した走行データから構成されているデータセットである．Caltech Pedestrian Dataset は，表 1 のように比較的規模が大きいデータセットであり，一般道や高速，トンネルなどの様々な場所で撮影された約 6 時間程の走行映像から構築されている．データセット中には，歩行者が約 35 万枚存在している．また，歩行者にはオクルージョンが発生しているか否かのラベルも付与されている．Caltech Pedestrian Dataset の評価では，このオクルージョンの有無と歩行者のスケールで評価の難易度を変更することができる．このように，データ数やデータに対するラベル情報，評価ツールが充実しているため，主要な歩行者検出のデータセットの 1 つとして用いられている．また，Caltech Pedestrian Dataset では，Receiver Operating Characteristic (ROC) カーブにより歩行者検出性能を評価している．

3.5　KITTI Vision Benchmark Suite [34]

KITTI Vision Benchmark Suite は，歩行者検出や車輌検出，道路認識等の Intelligent Transport Systems (ITS) 関係の研究向けに作成されたデータセットである．表 1 に示すように，KITTI Vision Benchmark Suite の学習サンプル数の規模は他のデータセットと比べて非常に大きい．KITTI Vision Benchmark Suite は数多くの都市を長時間撮影することで，多様な走行環境に対応したデータセットの構築を目的としている．また，RGB 画像だけでなく，ステレオ画像や LIDAR の 3 次元点群データ，GPS の地図情報などのデータも公開されている．これらのデータには，2 次元上の歩行者位置だけでなく，自動車の位置やこれらの 3 次元上の位置情報，及び地図上の位置のラベルも付与されている．このように，KITTI Vision Benchmark Suite はデータ数が非常に多く，ステレオ画像や LIDAR の 3 次元点群の情報など様々なデータが公開されていることから，注目されている．



(a) 前段処理による歩行者候補検出　　(b) CNN による歩行者検出

図 3　2 段階の検出構造による歩行者検出

表 2　CNN ベースの歩行者検出法の分類

	手法	年代	Miss rate	速度 (fps)	パーツ	カスケード	CNN特徴+識別器	スケールの対応	ネットワークモデル
2段階の検出構造	Joint Deep Learning	2013	39.3		✔		✔		CNN + RBM
	SDN	2014	40.63	0.7	✔				CNN + RBM
	EIN	2015	37.77	1					CNN
	TACNN	2015	34.99						AlexNet
	CCF	2015	17.32				✔		VGG
	Deep Cascade	2015	26.21	15		✔			VGG
	DeepParts	2015	11.89		✔				GoogLeNet
	CompACT	2015	11.75	2		✔	✔		CNN, VGGNet
Region proposal	Fast R-CNN	2015	12.86	3					Fast R-CNN
	SA-FAST R-CNN	2015	9.68	2.5				✔	Fast R-CNN
	Faster R-CNN	2015	18.02	2					RPN
	MS-CNN	2016	10	2.5				✔	RPN
	RPN+BF	2016	9.6	2			✔	✔	RPN
	SSD	2016	13.06	10				✔	SSD
	Fused DNN	2016	8.2	0.5		✔		✔	SSD + FCN

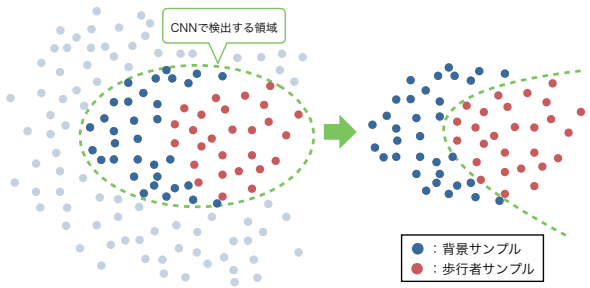


図 4　2 段階の検出構造の効果

4.　2 段階構造による歩行者検出

CNN を歩行者検出に応用する場合，ACF などの識別器と CNN を 2 段階に配置して歩行者を検出する．2 段階の検出構造による歩行者検出のアルゴリズムを図 3 に示す．前段の識別器では，入力フレームから歩行者の候補領域を検出する．ここで，前段の識別処理で歩行者の未検出を防ぐために，検出時のしきい値を下げる等の工夫が必要となる．そして，検出した歩行者領域を後段の CNN へ入力し，最終的な検出結果を出力する．

2 段階の検出構造は，図 4 のようなサンプルの分布が存在する場合，前段の識別器でサンプルの分布から CNN で識別する範囲を限定し，後段の CNN で詳細な識別を行っている．図 4 のように，2 段階の構造における後段の CNN で識別する対象を削減する効果があるため，誤検出を削減できる．

4.1　前段に用いる歩行者検出法

2 段階の検出構造において，前段で使用する識別器の選択は非常に重要である．もし前段の識別器で歩行者を取りこぼした場合，後段の CNN に入力されずに未検出となる．また，計算コストを考慮する場合，高速に処理が可能な識別器を用いる必要がある．2014 年に提案された Joint Deep Learning と SDN では，前段の識別器に CSS+HOG 特徴量を用いている．また，2015 年以降は，ACF-Caltech を前段の検出法として用いられている．本節では，前段の識別器として用いられている手法について詳細に述べる．

4.1.1　Color Self Similarity HOG [5]

CSS+HOG 特徴量は，勾配情報である HOG 特徴量にカラー情報である Color Self-Similarity (CSS) 特徴量を組み合わせた特徴量である．色情報は，歩行者の肌や服装，輝度の変化，人工物の色など色の変化が発生しやすいことから，歩行者検出に有効な特徴抽出が期待できないとされていた．しかし，CSS 特

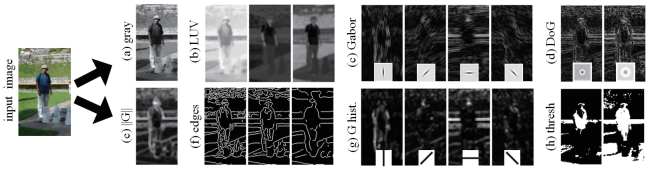


図 5　ICF 特徴量の抽出 (文献 [6] 引用)

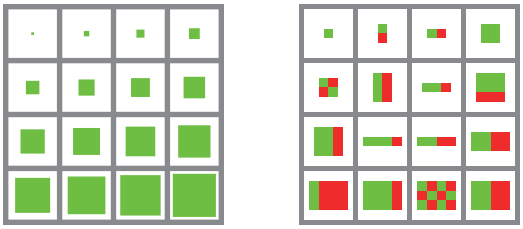


図 6　Filtered Channel Feature のフィルタパターン (文献 [35] 引用)

徴量は画像全体の色情報を用いるのではなく，画像内の 2 つの局所領域の類似度を特徴量として扱うことで，歩行者検出に有効な特徴量となる．HOG 特徴量をベースにした手法は，他にも数多く提案されており，文献 [1] のサーベイ論文にてまとめられているため，参考にして頂きたい．

4.1.2　Aggregate Channeled Feature

ACF は，ICF をベースにした手法であり，ICF のチャンネル特徴量を抽出した後に集約処理を施して特徴ピラミットを導入した手法である．ACF で用いるチャンネル特徴量は ICF と同様であり，図 5 の (a) 入力画像の輝度画像，(b) 表色系を変換した画像，(c) ガボールフィルタを畳み込んだ勾配画像，(d) Difference of Gaussian (DoG) フィルタを畳み込んだ画像，(e) 勾配強度画像，(f) エッジ画像，(g) 特定方向のガボールフィルタ，(h) 2 値画像の 16 チャンネルを特徴量を用いる．チャンネル特徴量を抽出した後，集約処理を行う．ACF の集約処理では，各チャンネル特徴量の局所領域内を総和し，新たなチャンネル特徴量を生成する．抽出したチャンネル特徴量は，特徴ベクトルに変換され，Boosted tree に入力される．Boosted tree に抽出したチャンネル特徴量を入力することで，膨大な特徴量群から歩行者検出に有効な特徴量を選択できる．

チャンネル特徴量をベースとした手法は，チャンネル特徴量の生成と Boosted tree による特徴量の選択が基本的な構造となっている．特に，チャンネル特徴量の生成において，バリエーションが豊富なチャンネル特徴量を生成することで，高精度化に繋ぐことができる．ACF の場合，集約処理により 1 枚あたりのチャンネル特徴量は縮小されているが，ピラミッドを作成することでチャンネル特徴量のバリエーションを増やしている．ACF をベースとした手法で，Locally Decorrelated Channel Features (LDCF) [8] と，SquaresChrFtrs [35]，Checkerboard [36] がある．これらの手法は，ACF の集約処理において局所領域内の総和を求めるのではなく，独自のフィルタパターンを畳み込むことで，チャンネル特徴量のバリエーションを増幅することで，より高精度な歩行者検出を実現している．

Locally Decorrelated Channel Features [8]

LDCF は，ACF の集約処理に学習サンプルの無相関フィルタを畳み込む処理を追加した手法である．集約処理で使用する無相関フィルタは，Linear Discriminant Analysis (LDA) [37] により求めている．これにより，抽出したチャンネル特徴量の相関を取り除くことができ，高精度化を実現することができる．

SquaresChrFtrs [35]

従来の ACF では，一定サイズのみで集約処理を行っていた．SquareChrFtrs は，図 6(a) のような複数のサイズで集約処理を行っている．これにより，集約処理で発生していたスケールの小さな歩行者の特徴量が消失する問題が半減され，スケールの小さな歩行者を高精度化に検出できる．

Checkerboard [36]

Checkerboard は，ACF の集約処理に図 6(b) のチェッカーボードパターンのフィルタを畳み込む手法である．チェッカーボードパターンのフィルタは全部で 132 パターン存在しており，各フィルタを畳み込むことでチャンネル特徴量のバリエーションを増やし，高精度化を実現している．

4.2　後段に用いる CNN ベースの歩行者検出法

後段に用いられている CNN ベースの歩行者検出法は，表 2 のようなグループ分けができる．本節では，2. 章で述べた 1) パーツベースの歩行者検出法と 2) CNN 特徴量を用いた歩行者検出法，3) カスケード構造を導入した歩行者検出法，4) どのグループにも属さないその他の歩行者検出法について詳細に述べる．

4.2.1　パーツベースの歩行者検出法

歩行者のパーツ情報を用いた歩行者検出の初期の手法として，Joint Deep Learning [13] がある．Joint Deep Learning は，CNN で歩行者に対する各パーツのスコアを出力し，RBM に抽出した特徴量とスコアを入力して人か背景かを識別する．入力画像を Joint Deep Learning へ入力した際に，頭部や腕等の歩行者の各パーツに対するスコアを出力する．ここで，出力層の前層には Deformation 層があり，前層の特徴マップから不要な応答値を取り除いている．そして，Deformation 層で取得した特徴マップを用いて，歩行者の各パーツに対するスコアを得る．RBM には，CNN の畳み込み層で得られる特徴マップと各パーツに対するスコアを入力する．入力する CNN の特徴マップは，特定の領域にマスクをかけた状態で入力する．1 段目の CNN で得られた歩行者の各パーツに対するスコアは，RBM の各層の重みとして用いられる．Joint Deep Learning は，歩行者のパーツ領域の特徴量で RBM を最適化しているため，姿勢の変化に頑健な歩行者検出を実現している．

Joint Deep Learning が提案された後に，歩行者の頭部や腕等の局所的なパーツではなく，歩行者の上半身や腰，下半身のパーツを用いた Switchable Deep Network (SDN) [14] が提案されている．SDN は，歩行者の上半身や腰，下半身のパーツ領域から識別に有効なパーツを選択しながら学習することで，高性能な歩行者検出を実現している．歩行者のパーツの選択は，Switchable 層によって行われる．Switchable 層は，畳み込み処理とプーリング処理により特徴マップを Spatial Pooling

によりパーツ毎に分割し，Switchable Restricted Boltzmann Machine (SRBN) により歩行者のパーツを選択している．

また，Joint Deep Learning や SDN のように歩行者の各パーツの特徴量を用いて検出するのではなく，学習時に歩行者の特定のパーツに対してペナルティを与える DeepParts [20] も提案されている．DeepParts は，前段の検出位置とラベル位置の重なり位置から複数の CNN を構築し，前段の検出位置とラベル位置が重なっていないパーツに対してペナルティを与えることで，高精度な歩行者検出を実現している．DeepParts の学習では，前段の検出位置とラベル位置のそれぞれで学習した CNN と，これらを統合した CNN の 3 つを使用する．このとき，統合した CNN の規模は統合前の CNN より大きく，前段の検出位置とラベル位置でズレが発生しているパーツに対してペナルティが与えられ，最終的な出力が得られる．

4.2.2 CNN 特徴量を用いた歩行者検出法

CNN の特徴量と統計的学習法を組み合わせた代表的な手法として，Convolutional Channel Feature (CCF) [18] がある．CCF は，ACF と CNN を組み合わせた歩行者検出法である．従来の ACF では，LUV 画像や勾配画像等の 10 チャンネルのチャンネル特徴量を Boosted tree に入力していた．CCF では，CNN から得られる特徴マップをチャンネル特徴量として入力する．CCF で使用するネットワークモデルは，AlexNet や VGGNet，GoogLeNet [38] で検証されているが，16 層の VGGNet が最も良いとされている．CCF は，16 層の VGGNet から得られる多量の特徴マップから，歩行者検出に有効な特徴量を Boosted tree で選択することができるため，高精度な歩行者検出を実現している．

4.2.3 カスケード構造の歩行者検出法

カスケード構造を取り入れている歩行者検出法は，主に高速化を目的に導入されている．カスケード構造を導入した代表的な歩行者検出法として，Deep cascade [19] がある．Deep Cascade は，2 段の検出構造で最も高速な歩行者検出法である．ここで，Deep Cascade の前段には VeryFast が採用されている．Deep Cascade では，大小の規模が異なるネットワークモデルを 2 つ用い，カスケード状に配置している．ここで，小規模なネットワーク (Tiny CNN) は VeryFast の後段に使用し，大規模なネットワークは小規模なネットワークの後段に配置する．最も後段にある大規模なネットワークは，VeryFast と Tiny CNN が歩行者と判定した場合のみ使用されるため，高速に歩行者を検出できる．

その後，Deep cascade のように複数の識別器をカスケード状に配置するのではなく，識別に使用する特徴量の選択をカスケード状に行う Complexity-Aware Cascade Training (CompACT) [21] が提案されている．CompACT は，CNN から得られる特徴量や Self-similarity (SS) [39]，Checkerboard，LDA を特徴量とし，AdaBoost で弱識別器として特徴量を選択する歩行者検出法である．SS や CB，LDA，CNN の特徴量を AdaBoost で学習した場合，精度を基準に特徴量を選択するため，CNN の特徴量が数多く選択される．そのため，SS や ACF などの比較的高速に特徴量を取得できる特徴量が選択されず，計算量が膨大になり，識別速度が大幅に上昇する．CompACT は，学習時に特徴量の計算コストも考慮させることで，SS や ACF などの比較的高速に特徴量をカスケード型 AdaBoost で選択されるようにする．これにより，性能を維持しつつ計算量を削減できるため，高性能かつ高速な歩行者検出を実現している．

4.2.4 その他の歩行者検出法

Ensemble Inference Networks [15] (EIN) は，評価時に学習した 1 つのネットワークから構成が異なる全結合層を複数生成し，それらの出力を統合して最終的な出力を求めることで高精度な歩行者検出を実現している．はじめに畳み込み層およびプーリング層の処理を行い，特徴マップを得る．生成した特徴マップは，全結合層に入力するために特徴ベクトルへ変換される．次に，学習したネットワークの全結合層をもとにランダムに選択したユニットの応答値を 0 にする全結合層を N 個用意する．そして，特徴ベクトルを生成した全結合層に入力して各クラスのスコアを求める．これにより，構造の異なる全結合層を通して得られた各クラスのスコアを，生成した全結合層数だけ求めることができる．各クラスに対する最終的な出力は，中央値や平均値，最大値から求める．ここでは，どの値の算出方法が適しているかは，問題設定ごとに決めることができるように一般化している．

Task Assistance-CNN [20] (TA-CNN) は，Multi-task Learning により歩行者と背景の属性を学習させることで高性能な歩行者検出を実現している歩行者検出法である．使用した歩行者の属性は，向きや自転車に乗っているか，性別などの 9 種類である．また，背景の属性は 4 つのセグメンテーションのデータセットから付与されており，空や木，建物など 8 種類ある．背景の属性は Semantic Tasks という方法により付与される．Semantic Tasks は，セグメンテーションのデータセットから Hard Negative を検出し，セグメンテーションのラベルと Hard Negative の検出位置を照らし合わせることで属性を付与している．学習時は，Task‐Constrained Deep Convolutional Network (TCDCN) による学習誤差の重み付けと，Task-wise early stopping による過学習の抑制を行っている [40]．TCDCN と Task-wise early stopping は，メインタスクの学習を補助するようにサブタスクを学習できるため，精度向上を可能にしている．

5. Region proposal ベースの歩行者検出法

2 段階の検出構造では，1 枚のフレームに対してラスタスキャンにより網羅的に歩行者を探索している．それに対して，Region proposal ベースの手法は 1 枚のフレームから歩行者の位置をピンポイントで探索している．また，Region proposal ベースの歩行者検出法は R-CNN ベースと Single shot ベースの手法に分割できる．

Region proposal ベースの歩行者検出法は，表 2 のようなグループ分けができる．Region proposal ベースの歩行者検出法はスケールの小さな歩行者を検出できるようなアプローチが用いられている．以下のに，Region proposal ベースの歩行者検

出法について述べる．

5.1 R-CNN ベースの手法

Region proposal をベースとした手法は，R-CNN をベースにしている．R-CNN は，Selective search で検出した物体候補領域を AlexNet (及び VGGNet) に入力して物体検出する手法である．Selective search は，セグメンテーションを利用した物体検出法であり，セグメンテーション情報を繰り返しグルーピングしていくことで物体候補を大まかにセグメンテーションし，物体候補領域を検出する手法である．本節では，Region proposal ベースの代表的な手法である Fast R-CNN と Faster R-CNN について説明し，これらを用いた歩行者検出法を述べる．

5.1.1 Fast R-CNN [26] と Faster R-CNN [27]

R-CNN は，Selective search で検出した物体候補領域に対して 225×225 にリサイズし，後段の CNN に入力している．そのため，Selective search で検出した物体候補領域の数だけ CNN で認識する必要があるため，非常に計算コストが高い．Fast R-CNN は，R-CNN で最も計算コストが高い畳み込み処理の回数を大幅に削減することで，計算コストを削減している．

Fast R-CNN の構造を，図 7(a) に示す．はじめに，入力画像に対して Selective search を施し，物体候補領域を検出する．このとき，画像全体に対して重みフィルタを畳み込み，特徴マップを得る．Selective search で検出した物体候補領域の位置を，畳み込み処理で得られた特徴マップ上に反映し，その領域の特徴マップを切り出す．切り出された特徴マップは，Region Of Interest (ROI) pooling を通して後段の全結合層のネットワーク (以下 サブネット) に入力される．ROI pooling は，設定したグリッドサイズに区切った領域に対してプーリングすることで，どのような入力サイズの特徴マップに対しても任意のサイズの特徴マップを得ることができる．このように，各物体候補領域に対して CNN を通す必要がないため，R-CNN より約 100 倍高速に物体検出することができる．

しかし，Fast R-CNN は前段の物体候補領域の検出処理と検出した物体領域を認識する処理で分割されている点である．そのため，総合的な計算コストは前段の検出器に左右されてしまい，もし前段の検出器の計算コストが高い場合に総合的な識別時間が長くなってしまう．Fast R-CNN の前段処理で用いられる Selective search は，物体候補領域を求める際に繰り返し superpixel で推定した領域の統合を行うため，非常に計算コストが高い．Faster R-CNN は，図 7(b) のように Fast R-CNN のネットワークモデルをベースとし，物体候補領域を RPN により行うモデルになっている．RPN を導入することで，物体候補領域の検出とその領域の認識を同時に行うことができ，Fast R-CNN より高速に物体検出することができる．

はじめに，Fast R-CNN と同様に入力画像に対して畳み込み処理を行い，特徴マップを得る．RPN では，得られた特徴マップに対して検出ウィンドウをラスタスキャンして物体検出をする．ラスタスキャンする際に，RPN ではアンカーという検出方法を導入している．アンカーは，図 8 のように注目領域を中心に k 個の決められた形の検出ウィンドウを当てはめていき，ラスタスキャンする方法である．アンカーにより指定した領域を RPN に入力し，物体らしさのスコアと入力画像上の検出座標を出力する．また，アンカーで指定した領域は Fast R-CNN のサブネットにも入力され，RPN で物体と判定された際に物体認識を行う．

5.1.2 Faster R-CNN を用いた歩行者検出法

Fast R-CNN は，歩行者検出にも応用されている [22]．Scale Aware Fast R-CNN では，畳み込み層の後に設置するサブネットを，大きい歩行者の検出に特化したサブネットと小さい歩行者の検出に特化したサブネットを用意する．そして，この大小 2 つのサブネットの出力値を最終的に統合することで，歩行者のスケールに頑健な検出を実現している．出力値の統合は，各サブネットの出力値に重みを付与することで統合している．重み値は，前段の ACF で検出した歩行者の高さから，シグモイド関数に従って重みを求めている．

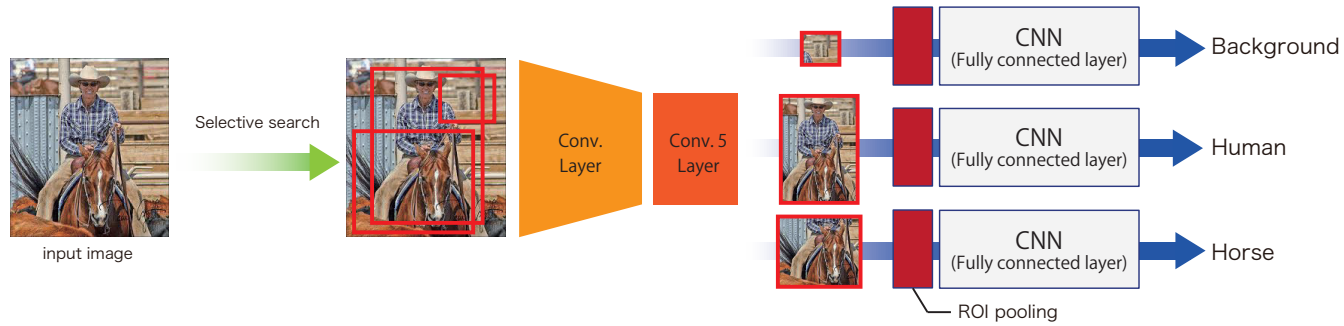
5.1.3 Region Proposal Network の応用 [21] [27]

Faster R-CNN の RPN は，歩行者検出にも応用されている．Cai らは，RPN のアンカーによる走査を 1 つの畳み込み層のみでなく複数の階層に適応することで，歩行者のスケールに頑健な歩行者検出を実現している [21]．CNN の入力層に近い畳み込み層はスケールの小さい歩行者の特徴を獲得し，出力層に近い畳み込み層はスケールの大きい歩行者の特徴を獲得しやすい．MS-CNN は，この CNN の特性を利用することで，1 つの RPN で複数スケールに対応したネットワークを構築している．

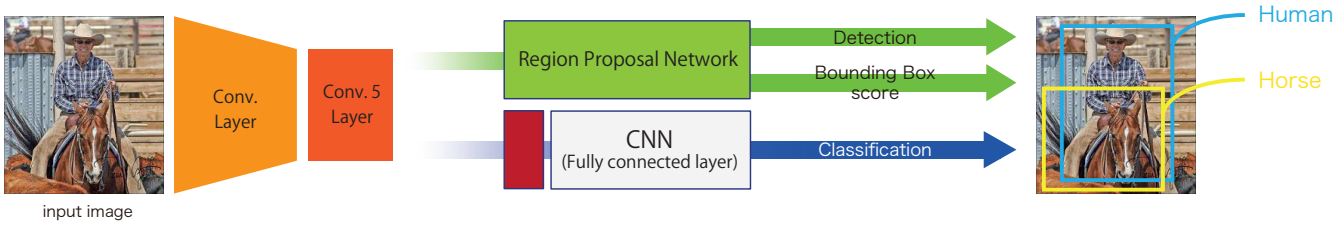
また，Zhang らは RPN と Boosted Forest を組み合わせることで，高精度な歩行者検出を実現している [27]．2 つ目は，歩行者に酷似した背景領域が多く存在していることである．Zhang らは，これらの問題を解決するために，特徴マップ生成の改良と Boosted Forest (BF) の導入を行っている．特徴マップ生成の改良では，小さいスケールの歩行者の特徴量を抽出しやすくするために，複数の層の特徴マップに ROI pooling を施している．従来の Faster R-CNN では，5 段階目の 3 層目の特徴マップに対してアンカーで走査をしている．しかし，プーリングの繰り返しにより，スケールの小さな歩行者の特徴が消滅してしまう．Zhang らの RPN は，4 段階目の 3 層目と 3 段階目の 3 層目の特徴マップに対してアンカーで走査をしている．このとき，ROI pooling を通した後の特徴マップのサイズを同一にするために，畳み込み時のスキップ幅を変更している．これにより，スケールの小さな歩行者に対して頑健な検出を実現している．Hard Negative に対しては，カスケード型の BF を用いることで対処している．BF を併用することで，歩行者を識別するために有効な特徴量を選択できるため，高精度化を実現している．BF には，検出ウィンドウの位置とその位置に対するスコア，特徴マップが入力される．

5.2 Single shot を用いた歩行者検出法

R-CNN ベースの代表的な手法である Faster R-CNN は，1 つの CNN で物体候補領域の検出と認識を行うことができるため，高速に物体検出できる．しかし，1 つのネットワークで物体候補領域の検出と物体認識を行っているが，これらの処理自体はサブネットが別になっているため，処理時間が余分に必要とな



(a) Fast R-CNN



(b) Faster R-CNN

図 7　R-CNN ベースの手法

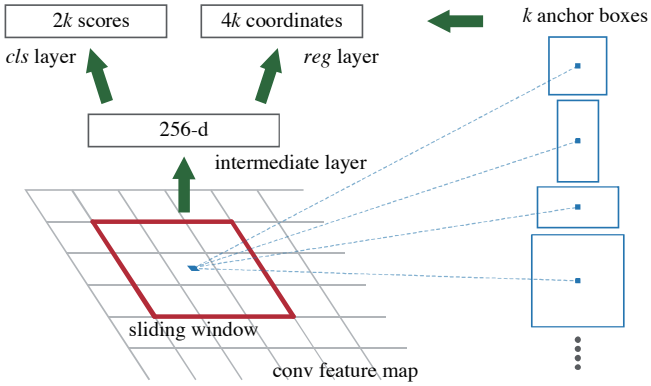


図 8　アンカーによる走査 (文献 [27] 引用)

る．Single shot ベースの手法は，1 つのネットワークで 1 回の処理で物体候補領域の検出と物体認識を行うため，R-CNN ベースの手法より高速に物体検出できる．本節では，Single shot ベースの代表的な手法である You Only Look Once (YOLO) と Single Shot Multi-box Detector (SSD) について説明する．

5.2.1　You Only Look Once [41]

YOLO による物体検出は，グリッドベースにより検出処理が行なわれる．まず，入力画像を指定したグリッドに分割する．ここでは，448 × 448 の画像に対して 7 × 7 に分割したグリッドを使用している．YOLO は，入力画像に対して各グリッドの物体のカテゴリと 2 つの BB の位置とスコアが出力される．そのため，出力層のユニットはカテゴリ数 (20 クラス) と 2 つの BB の位置とスコア ((x, y, w, h, スコア) × 2) を加算し，グリッド数 (7 × 7) を乗算した数となる．YOLO による物体検出は，各グリッド領域に対してカテゴリ分類と物体候補領域の検出をする構造になっている．

5.2.2　Single Shot MultiBox Detector [28]

YOLO は，大まかに定義したグリッドに沿って物体検出をしているため，物体のスケールの変化に弱い．Single Shot MultiBox Detector (SSD) は，図 9(b) のように各畳み込み層から物体候補領域と物体認識のスコアを出力させることで，物体のスケール変化を頑健にしている．SSD では，入力層に近い層でスケールの小さな物体候補を検出し，出力層に近い層ではスケールの大きい物体候補を検出している．入力層に近い特徴マップほど，プーリングによる特徴マップの縮小の影響を受けていないため，SSD ではこのような構造を採用している．SSD は，特徴マップの局所領域内毎に物体候補領域の矩形と認識結果を出力していく．そのため，Faster R-CNN のようにアンカー毎で矩形を出力する必要がなく，別のサブネットワークを介して物体認識をしなくても良いため，高速に物体検出をすることができる．また，SSD は End-to-End で学習できるネットワーク構造になっているため，高精度に物体検出できる．

SSD は歩行者検出にも応用されている [25]．SSD を歩行者検出に応用した Fused DNN は，SSD の歩行者検出結果とセマンティックセグメンテーションの結果を Soft-rejection based network Fusion (SNF) により統合することで，高精度な歩行者検出を実現している．Fused DNN では，はじめに SSD による歩行者候補領域の検出とセマンティックセグメンテーションを行う．ここで，セマンティックセグメンテーションには Fully Convolutional Network (FCN) をベースにした CNN を使用している．検出した歩行者候補領域とその領域のセマンティックセグメンテーションの結果は，SNF に入力されて最終的な歩行者検出結果を出力する．SNF は，検出した歩行者候補領域に対して 0 と 1 のラベルで学習するのではなく，SSD の検出スコアとセマンティックセグメンテーションの面積からラベルをスケーリングしている．使用するラベルが 0 か 1 の場合，学習時に誤識別した際に大きな誤差を出力するため補正が難しくなるが，SFN によりラベルのスケールを変更することで補正が容易

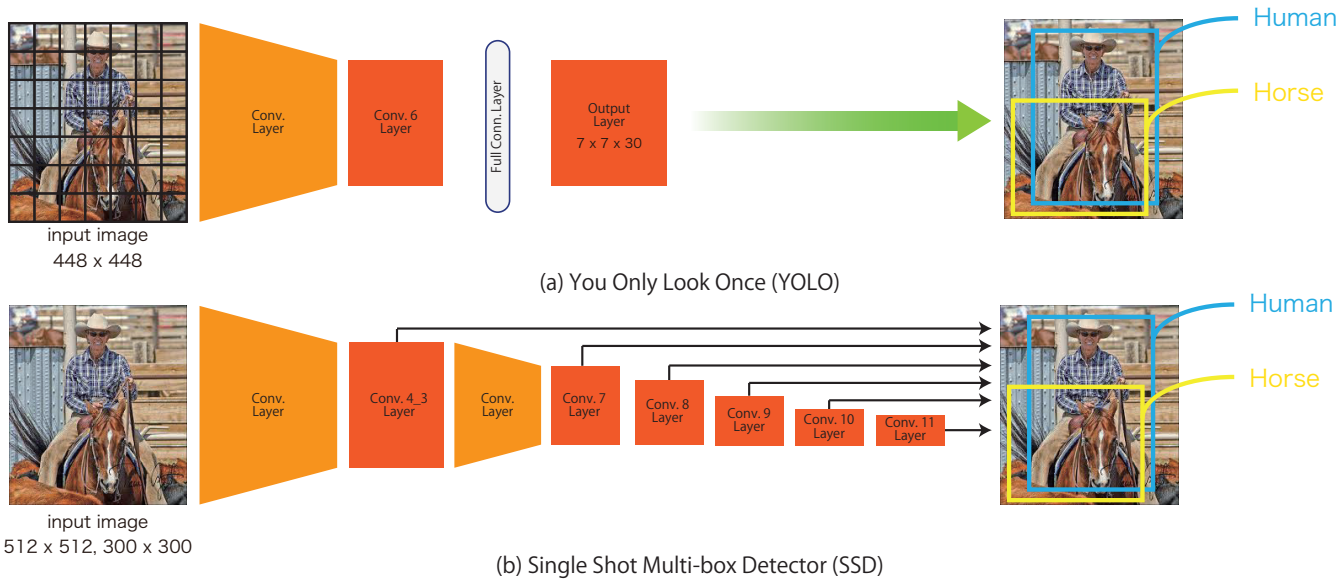


図 9　Single shot ベースの手法

になり，学習効率が良くなる．これにより，Fused DNN は高精度な歩行者検出精度を実現している．

6. おわりに

本稿では，Deep Learning をベースとした歩行者検出法と評価に用いられるデータセットについて述べた．まず，歩行者検出に用いられる特徴量の移り変わりをサーベイし，Deep Learning を歩行者検出に応用した際のアプローチを 3 つのグループに分割した．1 つ目は，2 段階の検出構造による方法である．前段の識別器により物体候補領域を検出し，後段の識別器により最終的な検出結果を出力する構造であり，計算コストと誤検出を削減する効果がある．2 つ目は，Region Proposal ベースの方法である．2 段階の検出構造を用いる必要がないため高速に歩行者を検出でき，局所領域内のみの学習ではなく，1 枚のシーン全体から歩行者の位置をできるため，高精度に検出できる．3 つ目は，Single shot ベースの方法である．CNN の畳み込み層の局所領域内から直接検出結果を出力していくことで，Region Proposal ベースの方法より高速な検出を実現している．

また，Deep Learning の進展により，大規模なデータセットが用いられるようになっている．2012 年までは 2,200 枚規模の INRIA Person Dataset が用いられ，2016 年には学習サンプルの多い Caltech Pedestrian Dataset や KITTI Dataset が用いられている．

文　献

[1] 山内悠嗣, et al., ”画像からの統計的学習手法に基づく人検出”, 電子情報通信学会論文誌, Vol. J96-D, No.9, pp. 2017-2040, 2013.

[2] N. Dalal, et al., ”Histograms of Oriented Gradients for Human Detection,” Computer Vision and Pattern Recognition, vol.1, pp.886-893, 2005.

[3] P. Felzenszwalb, et al., ”A Discriminatively Trained, Multi scaled, Deformable Part Model,” Computer Vision and Pattern Recognition, pp.1-8, 2008.

[4] Z. Cai, et al., ”An HOG-LBP human detector with partial occlusion handling,” International Conference on Computer Vision, pp.32-39, 2009.

[5] W. Stefan, et al., ”New Features and Insights for Pedestrian Detection,” Computer Vision and Pattern Recognition, 2010.

[6] P. Dollar, et al., ”Integral Channel Features,” British Machine Vision Conference, 2009.

[7] R. Benenson, et al., ”Pedestrian detection at 100 frames per second,” Computer Vision and Pattern Recognition, pp.2903-2910, 2012.

[8] W. Nam, et al., ”Local Decorrelation For Improved Pedestrian Detection,” Neural Information Processing Systems,pp.1-9, 2014.

[9] P. Dollar, et al., ”Fast feature pyramids for object detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.36, pp.1532-1545, 2014.

[10] R. Olga, et al., ”ImageNet Large Scale Visual Recognition Challenge,” International Journal of Computer Vision, vol.115, pp.211-252, 2015.

[11] A. Krizhevsky, et al., ”Imagenet classication with deep convolutional neural networks,” Advances in Neural Information Processing Systems 25, eds. by F. Pereira, et al., pp.1097-1105, Curran Associates, Inc., 2012.

[12] P. Sermanet, et al., ”Pedestrian detection with unsupervised multi-stage feature learning,” Computer Vision and Pattern Recognition, pp.3626-3633, 2013.

[13] W. Ouyang, et al., ”Joint deep learning for pedestrian detection,” International Conference on Computer Vision, pp.2056-2063, 2013.

[14] P. Luo, et al., ”Switchable Deep Network for Pedestrian Detection,” 2014.

[15] H. Fukui, et al., ”Pedestrian Detection Based on Deep Convolutional Neural Network with Ensemble Inference Network,” IEEE Intelligent Vehicle Symposium, 2015.

[16] J. Hosang, et al., ”Taking a Deeper Look at Pedestrians,” Computer Vision and Pattern Recognition, 2015.

[17] Y. Tian, et al., ”Pedestrian Detection aided by Deep Learning Semantic Tasks,” Computer Vision and Pattern Recognition, 2015.

[18] B. Yang, et al., ”Convolutional Channel Features: Tailoring CNN to Diverse Tasks,” International Conference on Computer Vision, 2015.

[19] A. Angelova, et al., ”Real-Time Pedestrian Detection With Deep Network Cascades,” British Machine Vision Confer-

ence, pp.1-12, 2015.
[20] Y. Tian, et al., "Deep Learning Strong Parts for Pedestrian Detection," International Conference on Computer Vision, pp.1904-1912, 2015.
[21] Z. Cai, et al., "Learning Complexity-Aware Cascades for Deep Pedestrian Detection," International Conference on Computer Vision, pp.3361-3369, 2015.
[22] J. Li, et al., "Scale-aware Fast R-CNN for Pedestrian Detection," European Conference on Computer Vision, pp.1-8, 2015.
[23] Z. Cai, et al., "A Unied Multi-scale Deep Convolutional Neural Network for Fast Object Detection," European Conference on Computer Vision, pp.1-16, 2016.
[24] S. Zhang, et al., "How Far are We from Solving Pedestrian Detection?," Computer Vision and Pattern Recognition, 2016.
[25] X. Du, et al., "Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection," arXiv, no.1610.03466, 2016.
[26] R. Girshick, "Fast R-CNN," International Conference on Computer Vision, 2015.
[27] S. Ren, et al., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," Neural Information Processing Systems, pp.1-10, 2015.
[28] W. Liu, et al., "SSD : Single Shot MultiBox Detector," European Conference on Computer Vision, pp.1-15, 2016.
[29] C. Papageorgiou, et al., "A Trainable System for Object Detection," Int. J. Comput. Vision, vol.38, pp.15-33, 2000.
[30] M. Jones, et al., "Fast Multi-View Face Detection," Mitsubishi Electric Research Lab Technical Report, 2003.
[31] P. Dollar, et al., "Pedestrian detection: An evaluation of the state of the art," IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.1-20, 2012.
[32] A. Ess, et al., "Robust multiperson tracking from a mobile platform," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.31, pp.1831-1846, 2009.
[33] M. Enzweiler, et al., "Monocular pedestrian detection: Survey and experiments," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.31, pp.2179-2195, 2009.
[34] A. Geiger, et al., "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," Computer Vision and Pattern Recognition), 2012.
[35] R. Benenson, et al., "Ten Years of Pedestrian Detection, What Have We Learned?," European Conference on Computer Vision, 2014.
[36] S. Zhang, et al., "Filtered Channel Features for Pedestrian Detection," Computer Vision and Pattern Recognition, pp.1751-1760, 2015.
[37] B. Hariharan, et al., "Discriminative Decorrelation for Clustering and Classification," European Conference on Computer Vision, pp.459-472, 2012.
[38] C. Szegedy, et al., "Going deeper with convolutions," Computer Vision and Pattern Recognition, pp.1-9, 2015.
[39] S. Eli, et al., "Matching Local Self-Similarities across Images and Videos," Computer Vision and Pattern Recognition, 2007.
[40] Z. Zhang, et al., "Facial Landmark Detection by Deep Multi-task Learning," European Conference on Computer Vision, 2014.
[41] J. Redmon, et al., "You only look once: Unied, real-time object detection," Computer Vision and Pattern Recognition, 2015.