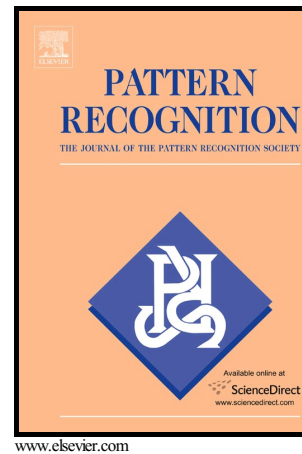


Author's Accepted Manuscript

Human Detection from Images and Videos: A Survey

Duc Thanh Nguyen, Wanqing Li, Philip O. Ogunbona



PII: S0031-3203(15)00317-9
DOI: <http://dx.doi.org/10.1016/j.patcog.2015.08.027>
Reference: PR5506

To appear in: *Pattern Recognition*

Received date: 10 January 2015
Revised date: 14 July 2015
Accepted date: 30 August 2015

Cite this article as: Duc Thanh Nguyen, Wanqing Li and Philip O. Ogunbona Human Detection from Images and Videos: A Survey, *Pattern Recognition*, <http://dx.doi.org/10.1016/j.patcog.2015.08.027>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting galley proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Human Detection from Images and Videos: A Survey

Duc Thanh Nguyen^{a,*}, Wanqing Li^a, Philip O. Ogunbona^a

^a*School of Computer Science and Software Engineering,
University of Wollongong, NSW 2522
Australia*

Abstract

The problem of human detection is to automatically locate people in an image or video sequence and has been actively researched in the past decade. This article aims to provide a comprehensive survey on the recent development and challenges of human detection. Different from previous surveys, this survey is organised in the thread of human object descriptors. This approach has advantages in providing a thorough analysis of the state-of-the-art human detection methods and a guide to the selection of appropriate methods in practical applications. In addition, challenges such as occlusion and real-time human detection are analysed. The commonly used evaluation of human detection methods such as the datasets, tools, and performance measures are presented and future research directions are highlighted.

Keywords:

Human detection, human description, object detection, object description

1. Introduction

The problem of detecting humans can be simply stated as: *given an image or video sequence, localise all subjects that are human*. This problem corresponds to determining regions, typically the smallest rectangular bounding boxes, in the image or video sequence that enclose humans. Figure 1 shows some examples of human detection.

During the last decade, human detection has attracted considerable attention in computer vision and pattern recognition largely due to the variety of applications it enables. In visual content management, a typical task is to tag (or label) the objects, especially humans, in the images and videos. Such tagging will enable subsequent annotation, search, and retrieval. Human detection is an essential component of automatic tagging. In video-based surveillance, one of the key tasks is to detect, identify, and monitor humans in crowded and public scenes such as airports, train stations, and supermarkets. Human detection is also found to be crucial in autonomous vehicles. It detects the presence of pedestrians on streets to alert the driver of dangerous situations. Examples include the ARGO vehicle¹ developed by the University of Parma and the Chamfer system² released by the University of Amsterdam and Daimler Chrysler. Recently, Mobileye³ launched the first vision-based collision warning system with full auto brake and

*Corresponding Author. Tel: +61 2 4221 3103, Fax: +61 2 4221 4170

Email addresses: dtn156@uowmail.edu.au (Duc Thanh Nguyen), wanqing@uow.edu.au (Wanqing Li), philipo@uow.edu.au (Philip O. Ogunbona)

¹<http://www.argo.ce.unipr.it/argo/english/index.html>

²http://www.gavrila.net/Research/Chamfer_System/chamfer_system.html

³<http://mobileye.com/>

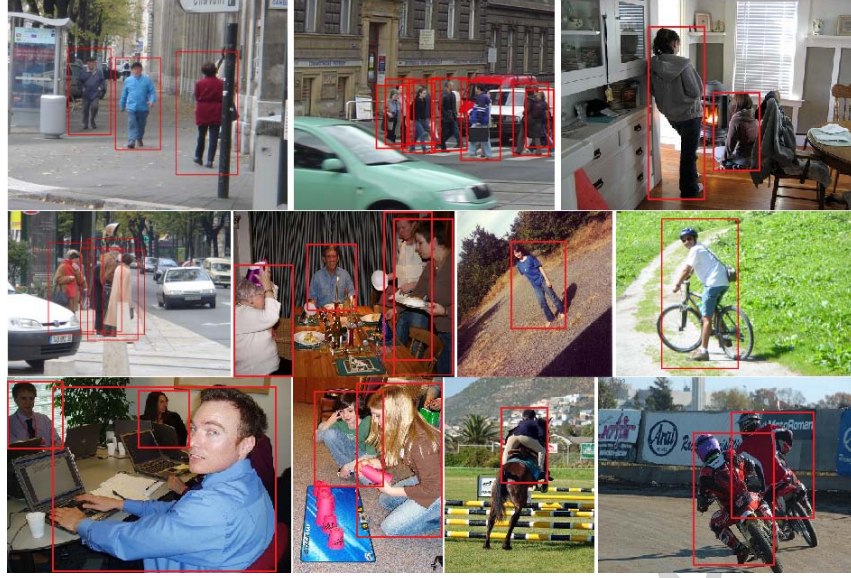


Figure 1: Some examples of human detection results.

pedestrian detection for use in the Volvo S60 cars. Figure 2 illustrates various applications of human detection.

Human beings are capable of detecting humans by only using subtle clues. Automated algorithms are instead still far from matching or even just approaching this ability. This is, in part, due to the intrinsic difficulties associated with the human body and the environment in which it is located. The non-rigid nature of the human body produces numerous possible poses. It is also challenging to model simultaneously view (orientation) and size variations arisen from the change of the position and direction (e.g. tilt angle) of the camera. Unlike other types of objects, humans can be clothed with varying colours and texture, which adds another dimension of complexity. In addition, the environment can make humans less visually noticeable. For example, the ambient illumination could either enhance or degrade the visual appearance of human objects. A cluttered background, often encountered in outdoor scenes, could camouflage humans. Furthermore, whether a single human is in activities or multiple humans interact with each other in crowded scenes, occlusions where the body of the human object is not completely observed are inevitable.

Human detection has been studied and progressed substantially in the past. The review of human detection has also been conducted in several works such as [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. However, our survey is different from those in the following dimensions

- Human detection in existing surveys was reviewed so far either as a component of a human motion analysis system, e.g. [1], [2], [3], [4], [5], or in the context of specific applications, for instance, pedestrian protection in a driving assistance system [6], [7], [8], [9], [10]. In contrast, this paper considers human detection as an object detection problem with an emphasis on the specific challenges posed by the articular nature and versatile visual appearance of the human body.
- Existing surveys decompose a human detection method into two components: features and classifiers. However, we have found that given the same feature, different ways of constructing the object descriptor from the feature could gain different performance. In addition, this scheme cannot well

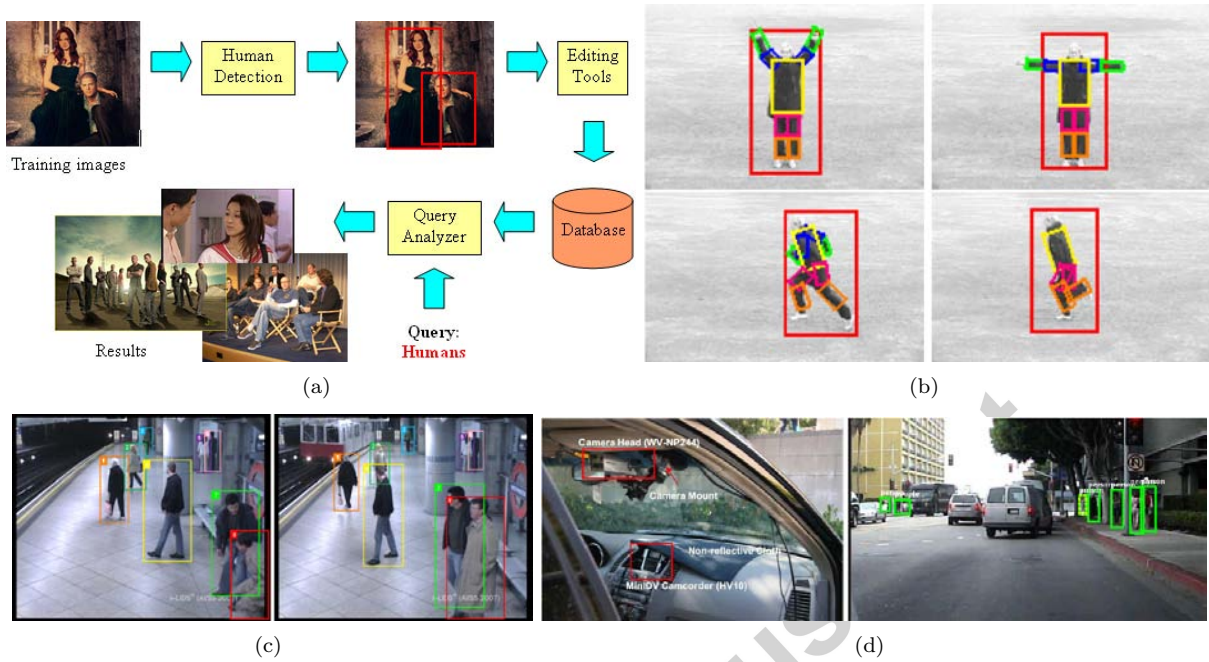


Figure 2: Applications of human detection. (a) Image/video retrieval. (b) Human activity recognition. Different body parts are labelled with different colours. (c) Surveillance systems (with detection and tracking). Humans are labelled using different colours. (d) Driving assistance system (from [9]).

explain the recent developments of human detection, e.g. in [11], [12], [13], [14], the structure of human objects was modelled to cope with deformations. Our attention, on the other hand, is on how to effectively describe human objects using features, i.e. human description, which plays a key role in the success of human detection. This viewpoint provides an insight into the state-of-the-art human detection algorithms and their suitability for various applications.

- Some challenges in human detection such as occlusion, real-time detection, and other recent developments such as use of context information and fine-grained human detection are presented and discussed comprehensively. We note that occlusion is also mentioned in previous human detection studies, e.g. [7], [9]. However, the occlusion was not well-defined and occlusion handling techniques were not considered specifically. This is probably due to the lack of advances and/or maturity of the approaches available at the time of the surveys. Indeed, there were few detection methods considering occlusion prior to 2010. In [9], an evaluation of several pedestrian detection methods under occlusion was conducted. However, most of the methods studied in this work were not intentionally designed to deal with occlusion. Thus, the experiment only served as an indicator of the sensitivity of detection methods to occlusion without in-depth analysis specific to occlusion. In our survey, the occlusion is firstly categorised and approaches to addressing the problem are reviewed and analysed accordingly.
- In addition to reviewing existing human detection approaches and common datasets, we also identify issues of current performance indicators and provide useful recommendations for potential applications.
- This survey covers most significant advances reported in the literature between 2010 and 2014.

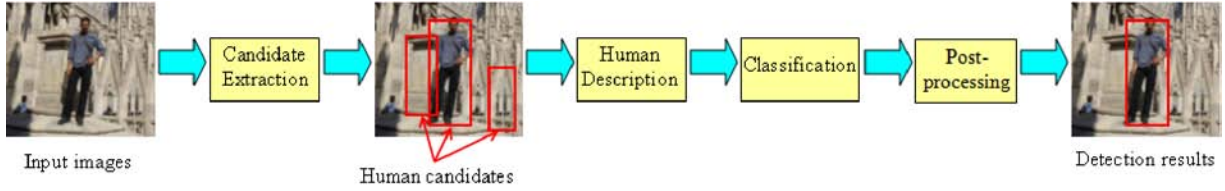


Figure 3: A general human detection framework. Note that preprocessing, e.g. image filtering, may be applied before candidate extraction to enhance the quality of the input image/video sequence.

The remainder of the paper is organised as follows. Section 2 describes a general human detection framework. The factors of a human descriptor are presented in Section 3. Those factors guide us to review existing human detection approaches. Other aspects of human detection such as classification and learning, detection under occlusion, reduction of computational complexity, and use of context information are presented in Section 4. Section 5 summaries existing datasets, tools, performance evaluation, and the issues related to the existing evaluation methods. Section 6 discusses some current trends and open issues for further research. Suggestions on how to choose a human detection algorithm for an application are presented in Section 7. Section 8 concludes the paper with remarks.

2. A General Framework for Human Detection

In general, the process of detecting human objects from images/videos can be performed in the following sequential steps: extracting candidate regions that are potentially covered by human objects, describing the extracted regions, classifying/verifying the regions as human or non-human, and postprocessing (e.g. merging the positive regions [15] or adjusting the size of those regions [16]). This process is illustrated in Figure 3. Note that this framework differs from that presented in [8]. In particular, foreground segmentation proposed in [8] is not assumed in our framework. This is because the segmentation step is not always required in existing human detection systems, e.g. [15]. Moreover, the tracking step in [8] is also not applicable for detecting humans from static images. Therefore, the framework proposed in [8] is specific to detecting pedestrians in videos captured by a static camera while the framework adopted in our survey is general, captures the recent developments of the field, and is applicable to a wide range of applications.

There are a number of ways to extract the candidate regions. A common approach is to assume that each human object can be enclosed by a detection “window”. Without any prior knowledge of the size and location of the human object, windows are extracted at various scales and positions. In this approach, it is expected that we need to merge some of the nearby windows that have been classified as human in order to obtain a final result. This is because that the human descriptor often tolerates some transformations, e.g. translation, scaling, of human objects in the detection windows. Figure 4(a) illustrates such a process, called window-based detection process. A commonly used method that merges multiple windows classified as human is the non-maximal suppression (e.g. [17]).

When the input to the detection system is a video sequence, a well-known technique, namely, background subtraction [18] can be used to obtain human candidates. In particular, moving objects are segregated from the background by calculating the difference between the current image and a reference background in a pixel-wise fashion. However, background subtraction often requires a static camera and



Figure 4: Various candidate extraction methods. (a) Window-based method (from left to right): input image with detection windows, windows that are classified as human, detection results after merging overlapping windows. (b) Background subtraction method (from left to right): background, input image, foreground objects. (c) Using stereo information (from left to right): left image, right image, depth map.

reference background containing no human to be given in advance. Some examples of this approach can be found in [19], [20].

If stereo images are available, depth information can be used to isolate human candidates [21], [22]. Knowledge of the ground plane can also be considered as an important cue to limit the search space of the location and scale of human objects in some methods, e.g. [23], [22], [24]. Figure 4(b)-(c) show examples of using background subtraction and stereo information to extract human candidates.

While extracting candidate regions is useful to enhance the efficiency of the detection by limiting the search space of human objects, the description of the human objects plays a key factor in the effectiveness and robustness of human detection. Indeed, when only static images or video sequences captured by a moving camera (e.g. in a moving vehicle) are given, neither background subtraction is applicable and nor depth information is available. In these cases, window-based detection process is the only possible approach and the performance of the detection very much depends on the human descriptor used.

3. Human Descriptors

In general, a human descriptor is comprised of features organised in a structure. It is expected that the structure enables the description of human objects in various viewpoints and poses. The features can capture the shape, appearance, or motion information of the human object. The features are computed at individual pixels or in local image regions. The local regions can be distributed in a dense structure (e.g. a regular grid [15]) or a sparse structure (e.g. points [16]). The most common approach to constructing a descriptor is to concatenate locally extracted features to form a high dimensional descriptor (or feature vector), e.g. [15]. In the following, both features and descriptors that have been developed for or employed

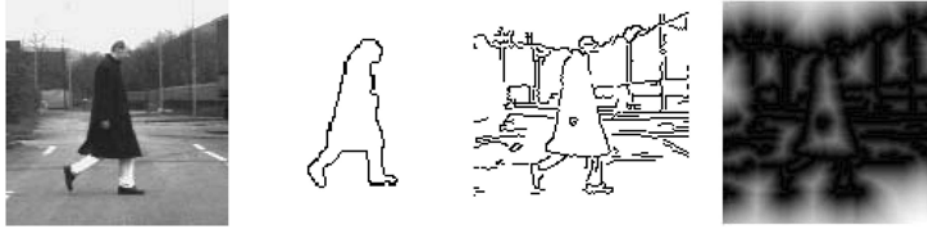


Figure 5: A conventional template matching process (from [27]). From left to right: Original image, edge template, edge map generated using some edge detector (e.g. Canny's detector [35]), and Chamfer distance transformed image. The gray value of each pixel in the distance transformed image represents the distance from that pixel to its closest edge. The higher the gray value is, the farther the distance is.

in human detection are reviewed. For the sake of brevity, the terms *object* and human object will be used interchangeably hereafter.

3.1. Features

In the literature of human detection, various features have been studied. These features can be computed from low-level information such as edge, texture, colour, or motion (when it is available). In this section, features are examined based on the visual characteristics they capture: shape, appearance, and motion.

3.1.1. Shape Features

To describe the shape of the human object, edge-based features are employed. The use of edge information in describing object shape has been verified not only in computer vision but also in psychological studies [25], [26]. In the edge-based features, the location, orientation, and/or magnitude of edge pixels are taken into account. The edge-based features can be extracted from either the edge maps or gradient images.

Pixel level edge-based features refer to the edge-based features computed at individual pixels. Examples of this approach include the methods in [27], [28], [29], [30], [19], [31], [32], [20]. In some methods, e.g. [27], [30], [19], [20], the location of each edge pixel can be encoded by its spatial distance to the nearest edge pixel on a template modelling the human shape. The templates can be as simple as parallel edge segments [29], rectangular contours [28], or small curves and segments called “edgelets” [32], [31]. However, to describe more complicated shapes, binary contours representing the human shape in various poses and viewpoints were used in [27], [30], [19]. Figure 6 shows some examples of edge templates. Calculation of the spatial distance between edge image and edge templates can be done efficiently by using the Chamfer distance transform [33]. This process is illustrated in Figure 5. Note that these methods are also called “template matching”-based methods in other surveys, e.g. [34].

The template matching approach has two major disadvantages. First, the pixel level edge-based features are easily interfered by noisy edges from cluttered background. Second, they are pose-specific. Thus, multiple templates are often required to cover various human poses. However, this leads to the computational burden. To overcome this problem, hierarchical template matching was proposed (see Figure 6). For example, Gavrilu [27], [30] organised the templates in a tree structure and the template matching was conducted only on few branches of the entire tree. In [19], the hierarchical structure was composed of three levels; each level contained templates representing a body part in various poses and

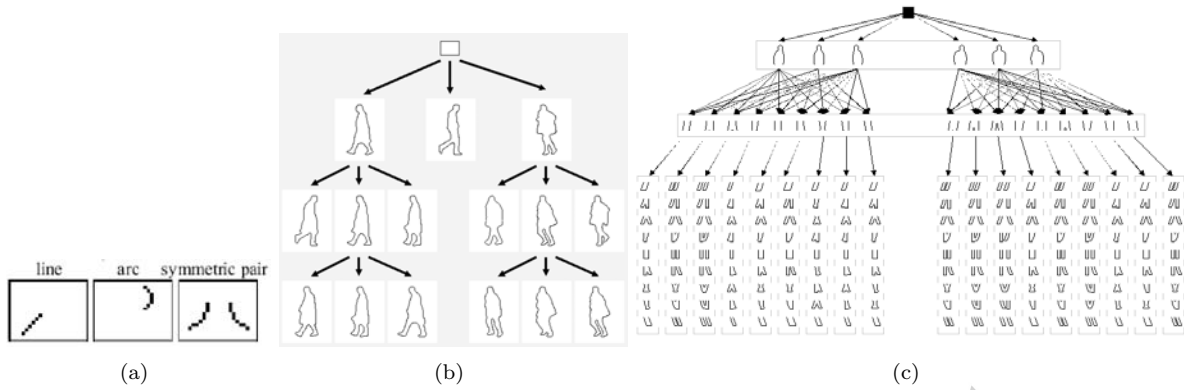


Figure 6: Some examples of edge templates. (a) Edgelets from [32]. (b) Full body templates from [27]. (c) Part templates from [19].

viewpoints. Based on the constraints on the human pose, the template matching was performed on parts by partially traversing the hierarchical structure.

Region level edge-based features. In contrast to computing edge-based features at pixel level, edge features obtained from local image regions have also been explored. Compared with the pixel level features, the region level features are to a certain extent adaptive to the local deformation of the human shape. A number of ways have been proposed to compute the region level edge-based features. For instance, the edge feature of a local image region is obtained by quantising the edge information of pixels in that region into discrete values and accumulated into a histogram of the quantised values. A well-known example of this manner is the histogram of oriented gradients (HOG) proposed by Dalal and Triggs [15]. The HOG was computed in a local rectangular region in which each edge pixel voted for a histogram bin corresponding to the edge orientation. The edge magnitudes were also used to weight the histogram bins. Figure 7(a) illustrates the HOG feature. The use of edge orientation had also been investigated previously in [36], [37].

Since it was proposed in 2005, HOG has been widely adopted and extended. For example, in [38], to obtain rotation invariant HOG, the dominant orientation of local regions was first estimated and then the HOG was computed relatively to the regions' estimated orientation. In [39], HOG features were computed at various quantisation levels of the edge orientation and non-rectangular regions were employed as complement to rectangular regions. In [40], the orientation of each pixel was evaluated at various scales of a Gaussian filter applied on that pixel. To reduce the dimension of the features, principle component analysis (PCA) was applied on the HOG in [11]. To make the human descriptor scale-invariant, in [41], the feature was formed by combining HOGs computed at multiple resolutions. In [42], the human image was first filtered using a Gabor filter bank. The HOG was then computed on the resulting Gabor image. In [43], various features were synthesized from the HOG to capture different properties of HOGs within a human object. For example, some synthesized features captured the spatial distribution of the HOGs extracted from local image regions. In [44], the difference of histogram of gradients (DHG) was proposed. The DHG was in fact a histogram in which each bin was computed as the absolute difference between two bins of the opposite directions in the original HOG.

Shape context proposed by Belongie et al. [45] for object recognition is also commonly used as a region level edge-based feature in human descriptors, e.g. [34], [46], [47]. Specifically, an image region was encoded by averaging the shape contexts computed at all edge pixels in the edge map of that image

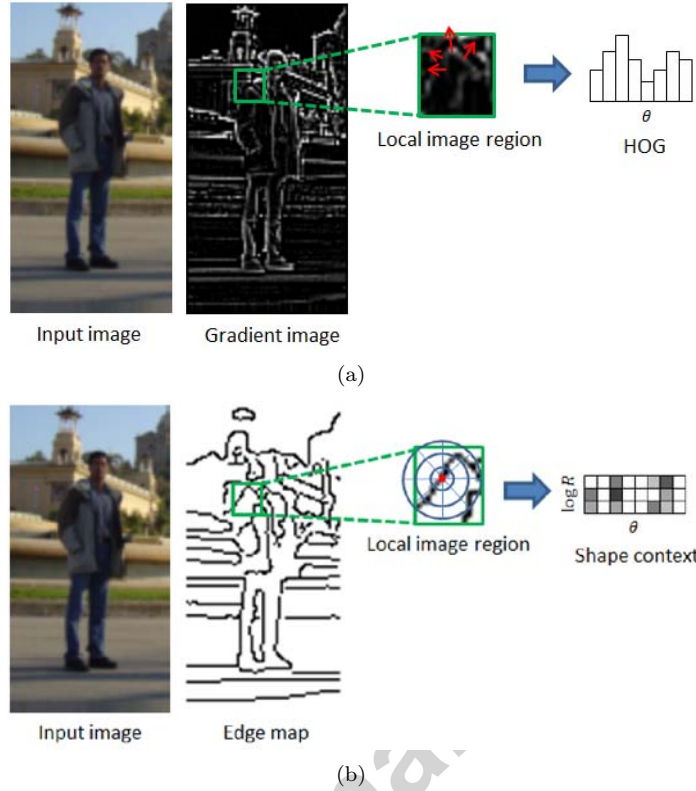


Figure 7: Shape features computed at local image regions. (a) HOG. Red arrows represent the gradients. (b) Shape context of a point (in red). R and θ represent the distance and orientation of pixels in a local image region in relative to that pixel.

region. The edge map can be obtained using some edge detector. Canny's detector [35] is a common one. Recall that the shape context of a point in a shape is the log-polar histogram of the distance and orientation of other points on that shape in relative to that point. In human descriptors, edge pixels of an image region are considered as points in a shape. Figure 7(b) illustrates the shape context feature.

Also computing edge-based features on local image regions, Sabzmeydani and Mori [48] proposed a so-called "shapelet" feature constructed from the oriented gradients. In particular, the shapelet of a local image region is the weighted sum of weak classifiers corresponding to individual oriented gradients of pixels in that image region. Adaboost algorithm is used to learn and select the weak classifiers. In [49], [50], a local image region was encoded by the covariance matrix of the spatial location, magnitude of gradients, and edge orientation of pixels in that region.

3.1.2. Appearance Features

Appearance features are mainly to capture the colour and texture and they are also extracted in local image regions. A simple appearance feature is the image intensity as adopted in [16], [23]. Haar feature is another commonly used appearance feature [51], [52], [53] (see Figure 8(a)). An extension of the Haar feature can be found in [54] where various configurations of the Haar feature were considered.

Local binary pattern (LBP) which was originally proposed for texture classification [55] was used to describe the appearance of the human body in [56], [57]. Similarly to HOG, an image region is encoded by the histogram of LBPs computed at all pixels in that region. LBP is well-known for its robustness against illumination changes, discriminative power, and computational simplicity. Many variants and extensions

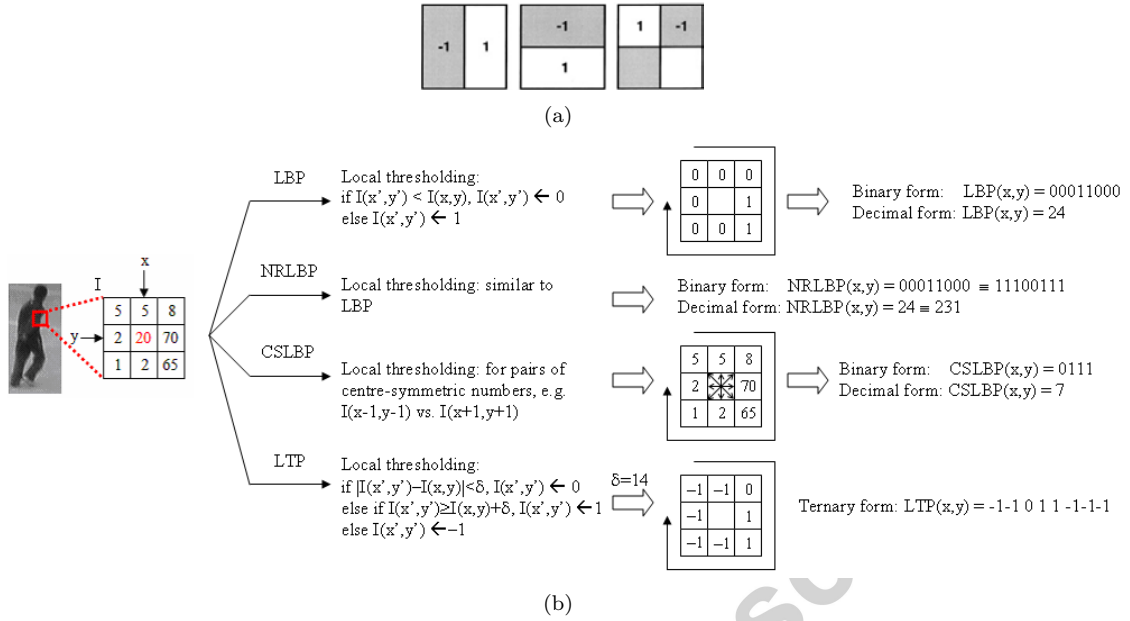


Figure 8: Some appearance features. (a) Haar features (from [51]). (b) LBP and its variants where (x, y) represents the location of a pixel on the image and $I(x, y)$ is the intensity of the pixel at that location.

of LBP have also been developed. For example, in [58], a variant of the LBP, namely Non-Redundant LBP (NRLBP) was proposed. The NRLBP considers a LBP code and its complement as same and thus reduces the number of possible LBP codes. However, as shown in [58], the NRLBP did not sacrifice the discriminative power compared with the original LBP. In [59], a texture feature was formed by combining the LBP [55] and NRLBP [58]. Center symmetric LBP (CSLBP) [60] obtained by comparing the intensity of symmetric pairs of local pixels were used in [61], [62], [63]. Local ternary pattern (LTP) proposed in [64] was employed in [57]. LTP is an extension of the LBP by allowing 3-valued quantisation of local intensity difference. As shown in [57], compared with the LBP, the LTP achieved better detection performance. In [65], LBP and LTP were generalised to a so-called “local intensity distribution” (LID) descriptor in which more quantisation levels were adopted and the neighbouring pixels in an LBP/LTP pattern were assumed to be independent. Figure 8(b) shows the LBP and its variants.

As shown in the literature, colour has also been considered as an important cue. For example, in [66] the HOG feature was computed on the segmented image obtained by segmenting an input image using the distribution of background and foreground colours. The feature therefore was called colour HOG (CHOG) which implicitly included colour information. In [67], the second order statistics of colour was proposed. This feature was calculated as colour self-similarity between the colour of pixels in different local image regions. In [57], LBP/LTP was computed on different colour channels. It was shown that computing the LBP/LTP on colour channels gained better performance than that on the intensity. However, the use of colour is application-dependent. This is because the availability of colour information is subject to the camera used in the application. In addition, the colour of human objects depends on human’s clothing which also varies.

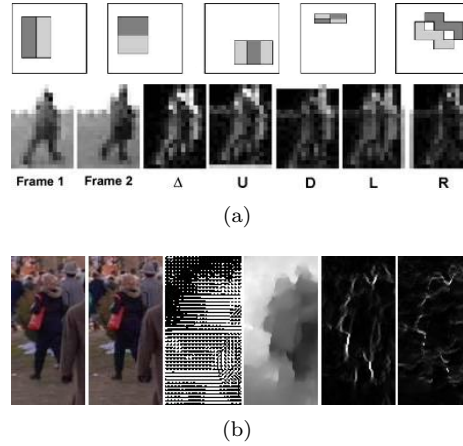


Figure 9: (a) Rectangular features (from [68]). First row: Feature filters. Second row: Temporal differences. From left to right: Frames I_t and I_{t+1} at time t and $t + 1$, $\Delta = \text{abs}(I_t - I_{t+1})$, $U = \text{abs}(I_t - I_{t+1} \uparrow)$, $D = \text{abs}(I_t - I_{t+1} \downarrow)$, $L = \text{abs}(I_t - I_{t+1} \leftarrow)$, $R = \text{abs}(I_t - I_{t+1} \rightarrow)$ where \uparrow , \downarrow , \leftarrow , and \rightarrow represent one-pixel image shifting operators. (b) HOF (from [69]). From left to right: two consecutive image frames, optical flow, flow magnitude, gradient magnitudes of the horizontal and vertical flow fields.

3.1.3. Motion Features

The availability of motion information can be exploited in human descriptors. Motion can be used to discriminate one object from another if the motion patterns are different and, thus, plays important role in object description. This is especially true for non-rigid objects such as the human body which often performs cyclic movements. To encode the motion of human objects, temporal features are defined from the temporal difference [68] or optical flows [69]. Similarly to shape and appearance features, motion features are also computed on image regions but those image regions are located on different frames.

In [68], rectangular features proposed in [70] were calculated on the difference images between consecutive frames to encode the temporal difference in the motion of pedestrians (see Figure 9(a)). In a similar way, Nguyen et al. [71] applied the NR-LBP proposed in [58] on the difference images as the motion feature. In [72], to alleviate the small amount of motion due to slow moving humans, the temporal difference was computed using multiple frames. Moreover, to obtain the part-centric motion information, e.g. the movement of a human's limb, image frames were stabilized (i.e. aligned) prior to computing the temporal difference. The stabilization of an image frame was performed by warping that frame using the optical flows extracted from the previous frame.

For the use of optical flows, histogram of flows (HOF) was proposed in [69]. The HOF was computed in a similar manner with the HOG [15]. The HOF can be used to describe the boundary motion as well as internal motion (i.e. the motion of internal regions of the human body). Figure 9(b) shows the HOF feature. In [73], the HOG was extracted on the flow images to encode the motion of pedestrians. It would be useful to confirm the presence of a human object if the body parts or joints can be tracked through motion. This idea was exploited in the work of Zhou et al. [74]. In particular, spatio-temporal patterns modelling the motion of joints were represented by trajectories and learned from the training data. Given a hypothesis of being a human, the trajectories were computed [75] based on tracking feature points using the dense optical flow method [76]. The hypothesis is then validated by finding a subset of those trajectories that best matched the spatio-temporal patterns learned off-line.

3.1.4. Combination of Features

Combining various cues could also improve the discriminative power of object descriptors; different types of features can supplement different information to the descriptors. For example, edge orientation histogram (EOH) proposed by Levi and Weiss [77] together with rectangular features [70] were used in [78]. In [79], three types of features: Edgelet [32], HOG [15], and covariance matrices [49] were combined. In [80], shape (represented by the contours and matched using Chamfer template matching) and appearance (encoded by image intensity as in [16]) were exploited. In [81], [82], HOG was integrated with LBP. As shown in [83], the combination of HOG and HOF outperformed the solely use of each feature type. In [66], the colour feature CHOG (described above) was combined with HOG. In [84], HOG and histograms of colours defined on the grayscale, RGB, HSV, and LUV channels were used. Similarly to [84], however only HOG and LUV channels were considered in [85]. In [67], the second order statistics of colours was proposed as an additional feature which was complementary to the HOG, HOF, and LBP. The HOG, HOF, and so-called HOS (i.e. HOG applied on stereo) were exploited in [86]. In [57], the combinations of HOG, LBP, and LTP were investigated. In [73], HOG was computed on the visual, stereo, and optical flow images. In [72], motion features (computed via temporal differencing), HOG, and histograms of colours [84] were aggregated. In [87], HOG, LBP, and LUV channels were used. In [14], Haar-like features, channels of gradients, and LUV were combined. A study on combination of features for pedestrian detection was conducted by Wojek and Schiele in [83].

3.1.5. Discussion

In summary, shape features, especially the HOG [15], show their preferences through many human detection studies in the literature during a decade. This is also consistent with findings in psychological studies [25], [26] indicating that shape features are discriminative to be used to discern objects in human perception. However, shape features are also well-known for being sensitive to clutter. For appearance features, texture features (e.g. the LBP [55]) are also informative. However, compared with shape features, appearance features are subject to the human's apparel which also varies. It is worthwhile to note that texture features (e.g. the LBP) are robust enough to be used solely (e.g. [56]) but colour features are often used to augment the appearance information in human descriptors (e.g. [66], [67], [87]). This is because colour has less discriminative power to be used independently, e.g. a human can wear a shirt of blue colour which is the colour of the sky. Motion features are important but the usability is limited to the cases in which the temporal information of the scene is available (e.g. the input is a video sequence) and/or the human object is in motion. Moreover, compared with other feature types, motion features are only meaningful when extracted on moving parts and thus the moving parts must not be occluded during the detection.

Although the combinations of various types of features have shown improvement compared with the use of individual features, there is still a lack of insightful explanation on the success of such combinations. Meanwhile the selection of features in the combinations is hand-crafted and the performance depends on the characteristics of evaluated datasets. For example, in some datasets colour patterns (of the human's clothing) may be prominent but they are totally dismissed in other sets. Table 1 summarises the feature types used in main human detection methods (from 2005 – 2014) in this survey.

Table 1: Summary on the feature types and descriptors used in human detection methods in this survey. The methods are sorted in their publication year (from 2005 – 2014).

	Shape		Appearance		Motion		Combined Features	
	Grid-based	Point-based	Grid-based	Point-based	Grid-based	Point-based	Grid-based	Point-based
2005	[15]	[34]		[16]			[68]	
2006		[88], [89], [90]			[69]		[91]	
2007	[38], [39], [48], [92] [93]	[19], [30], [31], [32] [46]	[54]	[23]			[78]	[22]
2008	[49], [94], [95]	[96]	[53], [56]				[79], [97]	[80]
2009	[98], [99]	[20], [47], [100]					[66], [81], [83], [84] [101], [102]	[101]
2010	[11], [12], [40], [41] [43], [50], [104], [105] [107]		[61], [62], [103]	[58]			[24], [57], [67], [73] [86], [106]	
2011	[108], [109]		[110]	[65]		[71]	[82]	
2012	[111], [112], [113], [114] [117]		[63], [115]				[116]	
2013	[42], [118], [119], [120] [126], [127], [128], [129] [13]		[59], [121]	[122], [123]			[72], [85], [124], [125]	
2014	[44], [130], [131], [132] [134], [135], [136], [137] [138]					[74]	[87], [14], [133], [134]	

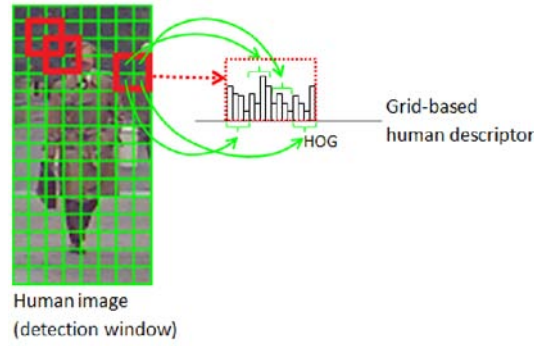


Figure 10: Human description using HOG feature and grid-based construction scheme proposed by Dalal et al. [15]. Red squares and green squares are the blocks and cells respectively.

3.2. Construction of Human Descriptors

Construction of human descriptors from features is equally as important as selection of the features. The descriptors are constructed by effectively combining the features from local regions. These local regions can be determined by sampling an object's image with a regular grid. Such an approach is referred to as *grid-based* construction. Alternatively, local regions can be located sparsely at a set of salient locations. This approach is referred to as *point-based* construction.

3.2.1. Grid-based Construction

The most popular method of this stream is the work in [15]. In this method, an object image is normalised to the size of 64×128 and uniformly divided into a dense grid of overlapping blocks. Each block is then split into 2×2 non-overlapping cells of size 8×8 pixels where histograms of oriented gradients (HOGs) are extracted. The object is then encoded into a feature vector created by concatenating the HOGs computed at cells and blocks. The size of the block can be fixed as in [15], [81] or varied as in [91], [54], [49], [56], [112]. The whole process of encoding a human object using HOG feature and grid-based construction scheme is shown in Figure 10.

A disadvantage of this approach is that the object descriptor computed in a regular grid of dense regions cannot adequately capture the actual object. Figure 11 illustrates such disadvantage. As can be seen in Figure 11, the second region (from left to right) in the first row of the grid contains a part of the head-shoulder pattern of a human object in the left image. However, the corresponding region contains none or less information on the object in the middle and right image respectively. Additionally, irrelevant information obtained at some locations on the grid, e.g. the background, is not informative and thus will corrupt the descriptor as well as make the descriptor redundant. To eliminate such irrelevant and redundant local regions, Adaboost-based learning method is often used to select local regions involved in the descriptor, e.g. [68], [91], [54], [49], [56], [112]. However, since the local regions are determined once through training, the descriptor is not adaptive to object deformation. To deal with this issue, Lin and Davis [105] proposed to extract local regions close to the training edge templates representing the human object in various poses and viewpoints. In a similar manner, Zhang et al. [14] simply defined templates as rectangular parts corresponding to the head, torso, and legs. Local regions were then extracted so that each local region contained more than one part type, i.e. on the borders between two parts or between a part and the background.



Figure 11: Illustration of the sensitivity of grid-based descriptor to object deformation (from [122]).

3.2.2. Point-based Construction

In the point-based construction approach, corners [139], [140] and scale-invariant points [141] are often considered as key points at which the local features are extracted. The local features are then matched with a predefined dictionary (or codebook) of typical feature patterns (called codewords) and the object descriptor is formed by the matched features. The most popular method of the point-based human descriptor is the work of Leibe et al. [16] in which the local features are represented as intensity patterns of local image patches extracted at the key points. In this method, each codeword not only represents its local visual characteristics but also implicitly encodes its spatial information. Other examples of the point-based construction approach include the works in [34], [88], [89], [90], [80], [74] in which various types of features were used to describe codewords. For example, shape features were used in [34], [88], [89], [90], [96], [100], [101], motion information was employed in [74], the combination of shape and appearance was exploited in [80]. In some methods, e.g. [96], [100], [101], histogram of the local features was used as the object descriptor.

The common issue of the point-based construction approach is that the discriminative power of the descriptor relies on the robustness of the key point detectors. However, existing key point detectors detect the key points locally and independently without considering the topology (i.e. the spatial relationship) between the key points. Thus, the key points may not necessarily represent and capture sufficient information about the object. Moreover, they are also sensitive to clutters. Figure 12 shows an example of scale-invariant key points. As shown in the three rightmost images on each row of this figure, many key points are detected in the background of the image while key points on the foreground object are missing.

To overcome this limitation, Nguyen et al. [122] proposed to locate the key points on the image edges that best capture the human shape using template matching. The templates are used to constrain the topology of key points to form a meaningful object and, at the same time, to filter out irrelevant key points belonging to the background or unimportant parts. Local features, e.g. NRLBP [58], were then computed on the local image regions centred at these points and concatenated to form a human descriptor. However, the robustness of such a human descriptor is scalable with the variability of templates used to represent possible poses and viewpoints of the human shape. The more templates used, the more robust the descriptor would be if the required computation can be accommodated. In addition, key point extraction is performed on every detection window and, hence, more computations are required, while conventional key points (e.g. [141]) are detected once on the whole input image.

General speaking, compared with the grid-based construction approach, point-based construction holds several advantages. Firstly, compact object descriptor can be created since only few points are considered in the descriptor. Secondly, the descriptor is more appropriate to describe non-rigid objects with high articulation such as humans wherein locations of the points can be adaptive to the variations of the object's viewpoints and poses.



Figure 12: Interest points detected using SIFT detector [141] (first row) and local image regions located at those interest points (second row). Size of patches is also determined using SIFT detector.

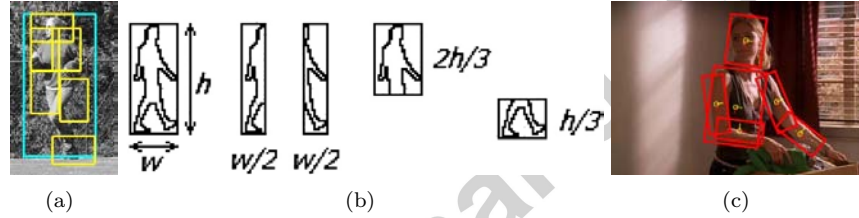


Figure 13: Some part models: Non-oriented parts in [95] (a) and [122] (b). Oriented parts in [47] (c).

3.2.3. Global vs Local Construction

Grid-based and point-based descriptors can be constructed either globally (holistically) or locally (i.e. part-based). The global approach constructs a descriptor for the entire object (e.g. [15], [57]) while the local approach uses part descriptors for different parts constituting the whole (e.g. [95], [99]). Note that the parts are not necessarily associated with semantic body parts of a human object (e.g. [99]). The parts can be described and then detected individually and independently in different processes (e.g. [36], [92], [32], [99], [12], [130]) or simultaneously with the full human body (e.g. [52], [37], [95], [105], [122], [135]). When the part-based approach is taken, the combination (configuration) of parts has to be verified so that they form a valid human body. Figure 13 shows some part models used in the literature.

The holistic approach is simple and convenient for the design and implementation while fits well to the description of human objects with less articulation such as pedestrians. The most popular and dominant method of this stream is the work of Dalal et al. [15], [17]. Although many holistic methods had been proposed (e.g. [51], [68]), the work in [15], [17] opened a new paradigm for human detection as it offered superior performance compared with previous methods (e.g. [52]). We notice that most of the human detection studies in the period between 2005 – 2010 followed this approach but mainly focused on different types of features and/or classifiers. The success of this approach has also been verified in recent pedestrian detection methods (e.g. [85], [14]) and on recent datasets, e.g. Caltech dataset [9] (see the details in section 5.1).

Compared with the holistic approach, the part-based approach has more advantages: part-based models have the capability of describing objects possessing high articulation and can potentially cope

with occlusions. However, there are two issues with the part-based approach. First, how to define the parts and the number of parts. Second, how to validate the configurations made from the parts in representing a meaningful object. Parts of a human object can be pre-defined using prior knowledge, e.g. head-shoulder, torso and legs were the parts in [32], [92], [105]. Parts can be represented by rectangular image patches which can be either non-oriented as in most methods or oriented (e.g. [142], [47]). In [127], parts are selected so that they contain sufficient information (i.e. not too small) but show the rigidity (i.e. not too large). It is also argued in [127] that small parts can be used to approximate small non-affine warps (e.g. perspective, foreshortening, rotation) with less computation (by using only translations). However, parts are not necessary to capture any anatomical body parts. Moreover, they may compose of many body parts and thus overlap each other when they are detected independently. For example, in [99], [12], a new notion called “poselet” was proposed to represent parts. A poselet is defined so that it is discriminative and representative for a part of the human object in some pose and viewpoint. Technically, the poselets are tightly clustered in both the feature space (represented by the HOG) and structure space (represented by the spatial distribution of key points, e.g. the joints in the body, eyes and nose in the human face). As shown in [99], [12], a poselet can be the half of a frontal view and a left shoulder or even the full body of a human object. In general, the design of parts depends on the level of information required by applications as well as the level of articulation of humans in the applications. For example, small and oriented rectangular parts better fit with body parts in high articulation and are often used to obtain detailed description of the human body. Meanwhile, non-oriented rectangular parts are adequate to less deformed configurations.

The configuration of parts can be validated using a set of constraints on the relative sizes and geometric relationship between the parts and the regularity and consistency in their viewpoints and poses. The geometric relationship between parts can be determined in several ways as follows. It can be defined using prior knowledge. For example, in [52], [32], [53], [98], [105], [122], [123], the locations of parts relative to the human body were predefined in advance. In [29], [28], [36], the geometric relationship between body parts was described using probabilistic models. A logical reasoning framework employing geometric rules was proposed for verifying the human body structure in [92]. The geometric relationship between parts can also be learned. For example, in [142], [47], the locations of parts were considered as hidden variables of a Markov Random Field model and learned using a standard maximum likelihood estimation algorithm. In [95], [11], [127], the placements of parts were encoded in latent variables of a latent SVM and learned by training the SVM. In [135], common patterns of combination of poselets were called k -poselets and determined from the training data by minimising the intra-variation of the relative locations of poselets in the patterns. These common patterns with the associated locations of poselets capture various configurations of the human body. In other methods, e.g. [53], [98], the parts are located based on their classification scores. For example, in [98], the locations of parts were determined iteratively by using a greedy method called seed-and-grow. In this method, seeds are the locations with high classification scores (of parts).

In addition to modelling the geometric relationship between parts, validation of the regularity of parts in the viewpoint and/or pose has also been addressed by the part-based detection methods. For example, in [88], [89], the regularity of parts was encoded using the implicit shape model proposed in [16]. Specifically, each part refers to the possible viewpoints and poses that can cover the part and this viewpoint/pose information is used to validate the configurations of detected parts. For example, it is impossible to have a valid human object whose one leg refers to a standing pose while the other leg

refers to a running pose. In [105], body parts including head-shoulder, upper legs, and lower legs, were determined sequentially in a hierarchical structure in which the part at one layer in the tree structure was made based on the parts obtained at previous layers. However, determining body parts in this manner might lead to accumulated errors, i.e., if one part is not identified correctly, the error propagates to the subsequent processes. In [12], the consistency of parts (i.e. poselets) sharing common keypoints (e.g. nose, eyes, mouth, and joints) were used to validate the presence of human objects. Pose and viewpoint validation was accomplished based on the pre-estimated co-occurrence frequency of parts in different poses and viewpoints in [123]. In [130], the human body was expressed in an ordered sequence of parts implicitly encoding the pose and viewpoint information. The parts were considered as letters in an alphabet whereas the poses were treated as words. The validation of parts was then converted to string matching in text recognition. However, modelling 2D or 3D human poses and viewpoints in 1D sequences may omit high-ordered important information of the body layout which cannot be captured by sequences, e.g. the spatial relationship and co-occurrence of two parts that are not adjacent in the 1D sequence.

4. Some Aspects Related to Human Detection

4.1. Classifiers and Learning Algorithms

As shown in Figure 3, once human descriptors are extracted from the candidate regions, the classification step is invoked to classify the candidate regions as human or non-human. The classification follows either generative or discriminative approach.

4.1.1. Generative Approach

The generative methods aim to construct a model, e.g. shape model [27], [30], [19], structure model [36], appearance and structure model [16], [103], of the object of interest. For instance, template matching-based methods make use of templates for modelling the shape of the full human body [27], [30], or body parts [19].

An example of structure models is the work of Mikolajczyk et al. [36] in which the classification of a human object was performed using a naive Bayesian classifier based on the locations of the human's parts and the classification scores of the parts.

Another seminal work is the so-called implicit shape model (ISM) proposed by Leibe et al. in [16]. In the ISM, the appearance of human objects is captured in a dictionary (or called codebook) of codewords. Each local feature extracted on a human candidate is matched with several codewords. The matched codewords vote for possible locations of the human object (e.g. the centre of the bounding box containing that human). The relative location and size of a human object given local votes are obtained through training. Note that the training/learning phase in the generative approach makes use of only training samples containing instances of the human object. The voting process results in a Hough space of votes. The classification score of a human object is then computed by summing votes in the Hough space. The ISM was applied with modifications and extensions in the later works. For example, in [88], besides the spatial voting, shape clusters representing the viewpoint of humans were also involved in the voting scheme, i.e. each matched codeword also voted for a shape cluster in which the codeword occurred in training. In [89], the method in [88] and the original ISM [16] were combined. However, a codeword could vote for different shape clusters sharing the codeword. In other words, this strategy allows the cross-articulation. In [99], [12], the ISM was applied for poselets instead of local features as in [16]. In

[103], the relationship between local features and object hypotheses was modelled in a graph in which the voting was represented as the edges (links). The verification of the existence of a human object was then formulated as a maximum-a-posteriori inference in the graph.

4.1.2. Discriminative Approach

Since the classification of candidate regions can be considered as a binary classification problem, discriminative methods have been used. Many robust learning methods and classifiers have been investigated along this thread of thinking. The common aims are to improve the performance of learning and classification algorithms.

SVMs are often used to classify the human and non-human descriptors by maximising the margin between these two classes [15], [69], [17]. Satpathy et al. [44] observed that the human class often distributes in a small space surrounded by the non-human class in the feature space. This is due to the diversity of the non-human class. Consequently, a linear SVM may not be able to discriminate the two classes. To deal with this problem, a hyper quadratic classifier using the minimum Mahalanobis distance [143] was used. In [118], a piecewise linear SVM (PL-SVM) made up of a set of linear SVMs was proposed. The PL-SVM is defined as the linear SVM whose the distance to the corresponding hyperplane is maximal (in comparison to the other linear SVMs in the set). Ideally, the intent of each linear SVM is to discriminate human objects in some pose against non-human objects and human objects of other poses. Thus, the positive training set used to train each linear SVM was determined via clustering in which each cluster was supposed to represent a group of human object samples in some pose. It is possible to employ various complimentary features that provide richer descriptors. However, this might lead to intractable learning process insofar as training is concerned. This problem was addressed in [102] where Partial Least Squares (PLS) analysis was used as a dimensionality reduction technique that extracted prototypical features from the set of training features and ensured efficient training. In [94], histogram intersection kernel SVM was introduced for fast training and classification. In [95], [11], a latent SVM was proposed to learn the locations of parts of an object. This work also introduced a method to tune the SVM's parameters without using the whole set of training data, thus improved the efficiency of the training process. To make the human descriptor insensitive to the scale factor, Yan et al. [128] proposed to learn resolution transformations which map features (e.g. HOG) from different resolutions to a common feature space in training a SVM classifier. In [138], a feature alignment method was proposed to normalise features deformed by articulation. In [43], features were synthesized from the HOG and then selected by using a feature prediction algorithm based on the weighting scheme of the linear SVM.

AdaBoost with cascade architecture was first time used for human detection in [68]. This framework was then adopted and extended extensively in the later works. For example, to improve the discriminative power of the cascade AdaBoost architecture, Chen et al. [78] added a meta-stage using the classification scores of stages in the cascade framework to exploit the inter-stage information. In [125], the cascade architecture was resembled the tree structure enabling the Random Forest ensemble [144]. In a tree of the forest, a node was associated with a descriptor and a weak classifier used in the cascade framework. In [112], the cascade framework was built from SVMs as weak classifiers. $L1$ -norm minimisation learning was proposed to effectively obtain the sparseness of the weight vectors in SVMs. Furthermore, to minimise the number of weak classifiers, integer programming was employed as a special case of the $L1$ -norm minimisation in the integer space. In [54], a mining feature algorithm was proposed to automatically select features and obviate the need for manual feature design. The learning algorithm was designed to

deal with the trade-off between the accuracy and computational complexity of selecting features. Discrete AdaBoost then was used to combine the mined features in the final classifier.

Neural Networks are also used as the human classifier. For example, feed forward neural networks with various feature types were investigated in [145]. Recently, deep neural networks have shown their potentials in human detection [114], [13], [134], [137]. For example, in [114], various combinations of body parts were represented by nodes in a deep belief network. In [13] sub tasks of human detection such as feature selection, object description, occlusion handling were organised into different layers of a deep convolutional neural network and the parameters of each layer were jointly learned through the network. In [134], the mixtures of parts were encoded in a deep architecture called switchable deep network. The network was able to infer (and switch to) the most appropriate mode of the mixtures. The robustness of the network was also verified by using different feature types, e.g. the HOG and LBP. However, in general the design of the network architecture is important but requires specific knowledge and experience.

Multiple Instance Learning. To adapt with the articulation of the human body, multiple instance learning (MIL) [146] was employed in [98]. In this method, for each human image, a bag of instances was generated by slightly translating the detection window covering the human image region. A bag of instances is classified as positive if at least one of its instances is classified as positive and vice versa. In [53], Dollár et al. proposed a so-called multiple component learning (MCL) method. In the proposed method, components were described by part descriptors and classified by part classifiers. The part classifiers were considered as weak classifiers in a boosting algorithm and trained using MIL.

Transferring Learning. It is observed that a good classifier validated on a training/validation set may get poor performance when it is applied on test sets from different sources. To deal with this issue, transferring learning approaches (i.e. transfer the knowledge from the training sets to the test sets) have been applied. The simplest manner is to directly extend the training set by including few labelled samples from the test set [110]. Note that a small amount of samples from the test set is not sufficient to fully train a classifier, but useful to augment and enrich the classifier. In [121], labelled samples common to both the training and test set were selected (using manifold learning) to train an AdaBoost classifier applied on the test set. In [109], [132], samples on the test set were not labelled in advance but could be collected automatically. The confidence scores, representing the commonality, of samples were then computed. Common samples with their corresponding confidence scores were used to train a confidence-encoded SVM which was then used for the classification on the test set. To extend the training set for the adaptation of a SVM classifier to various test domains, Xu et al. [136] made use of the multiple instance learning approach. In this method, each positive sample (i.e. instance) was considered as an initial ground-truth and the bag containing this ground-truth human was formed by getting other samples which spatially overlapped the ground-truth human object and got high classification scores. In [137], the deep architecture proposed in [13] was enriched with a so-called cluster layer for learning scene-specific visual patterns and augmented with distribution modelling for joint learning the classifier and reconstruction of features. The nodes in the cluster layer were determined by using an unsupervised clustering method. The reconstruction model aimed to identify important samples influencing both the source (i.e. training/validation) and target (i.e. test) domains. The smaller the reconstruction error a sample obtains via the deep model, the more important the sample is.

4.2. Detecting Humans under Occlusion

One of the most difficult challenges in human detection is occlusion. In general, the term “occlusion” is referred to as a phenomenon in which an object of interest is not fully visible. The features extracted

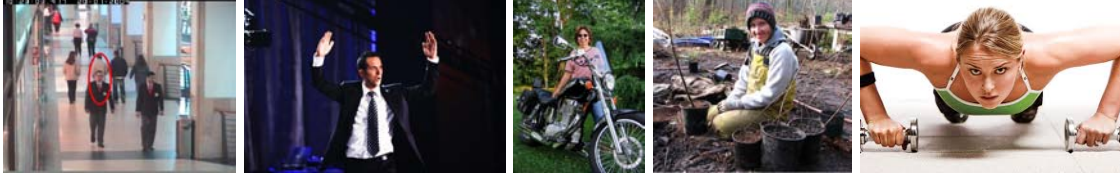


Figure 14: Occlusion types (from left to right): a human (marked in red) is occluded by another human, occlusion by image border, occlusion by a non-human object (motorbike), occlusion by pose (the lower legs are occluded due to the sitting pose), and occlusion by the camera's viewpoint.

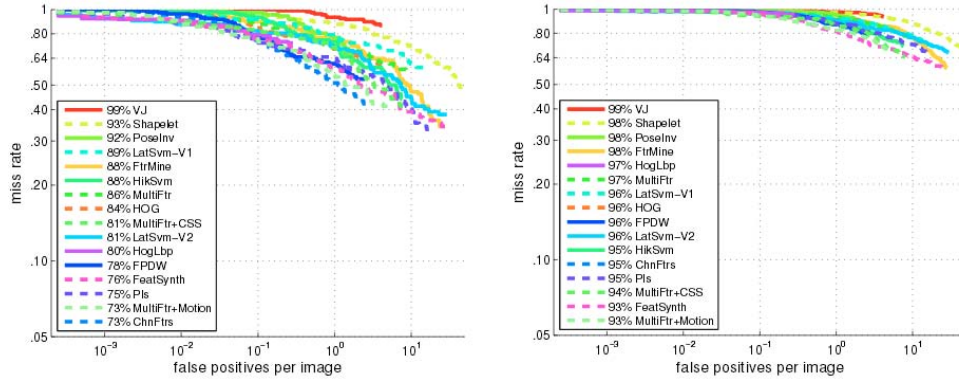


Figure 15: Detection performance of some pedestrian detection methods under occlusion. Left: occlusion level (i.e. area occluded) less than 35%. Right: occlusion level from 35% to 80%. Details of the performance measure are presented in section 5.1. Methods used in this comparison are aliased as VJ [68], HOG [15], Shapelet [48], FtrMine [54], HikSvm [94], LatSvm-V1 [95], MultiFtr [97], HogLbp [81], Pls [102], ChnFtrs [84], PoseInv [105], FPDW [106], MultiFtr+CSS [67], MultiFtr+Motion [67], LatSvm-V2 [11], FeatSynth [43]. This figure is from [9] and best viewed in colour.

on occluded portion of a human object would be corrupted and therefore bias the classifiers. In practice of human detection, occlusions can happen in the following scenarios. First, a human object is blocked by another object called occluder. The occluder can be either another human object, as often seen in surveillance applications, or a non-human object, e.g. a car occluding a pedestrian or even just an image border. We refer this type of occlusion as inter-object occlusion. Second, a human object may not be fully observed due to his/her pose and/or the camera's viewpoint. This type of occlusion is referred to as intra-object occlusion or self-occlusion. Figure 14 illustrates the occlusion types. In [9], a performance comparison of some pedestrian detection methods was conducted on the Caltech dataset (details of this set are provided in section 5.1). We note that this dataset was created specifically for pedestrian detection and thus includes inter-object occlusion cases, e.g. occlusions made by cars and/or trolleys. In addition, except the method [81], other detection algorithms studied in [9] were not intentionally designed for tackling occlusion. Figure 15 shows this comparison.

As shown in the literature, a number of methods have been proposed to address the occlusion problem. However, as far as we are aware, existing approaches mainly focus on the first type of occlusion, i.e. inter-object occlusion. The current occlusion handling methods can be categorised as detection-based occlusion handling or inference-based occlusion handling. The detection-based approach determines the occlusion of a human object by using only the information of that object and its parts, e.g. detection scores, geometric information of parts. Meanwhile, for inference-based approach, the occlusion of a human object is inferred based on the mutual relationship between that object and other objects.

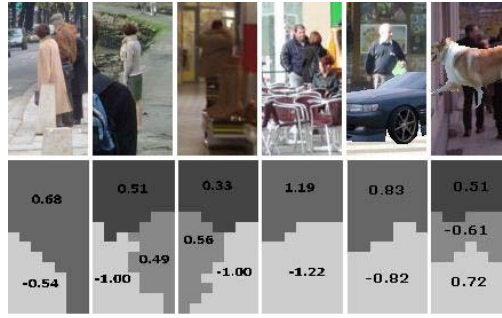


Figure 16: Some examples of partially occluded humans (first row) and segmented regions (second row) with responses (numbers) returned by a holistic linear SVM. The negative responses indicate possible occlusion. This figure is generated from [81].

4.2.1. Detection-based Occlusion Handling

For example, in [92], a logic based reasoning framework was proposed for occlusion handling. The framework used a number of logical rules based on the response of each individual part detector and the geometric relationship between the detected parts. However, the consistency in the poses and viewpoints of the parts was not taken account in this method.

In [81], grid-based descriptor with HOG feature and linear SVM [15] were used. Responses of the linear SVM classifier to HOG features computed on local regions of the grid were employed to construct an occlusion map for each human hypothesis. The occlusion map was then segmented using the mean-shift algorithm [147]. Segmented regions of mostly negative responses were considered as occluded regions while regions of positive responses (implied as non-occluded regions) were classified using part classifiers. There were only two types of part classifiers, upper-body and lower-body classifier, used in [81]. Figure 16 illustrates the method in [81] with several occlusion cases (first row) and corresponding classification responses (second row). In a similar manner to [81], however, in [131], part classifiers were determined using random subspace classifiers selection technique [148]. The final classifier was built as a linear combination of the selected part classifiers. A disadvantage of the above methods is that they are not applicable when non-grid based descriptors are employed. This is because the occlusion map was constructed from the dense representation of local regions in the grid. Moreover, the method in [81] is strongly coupled to the use of linear SVMs since the response of the classifier to each local region was gained conveniently from the formulation of a linear SVM classifier.

In [73], motion (optical flows) and depth (stereo vision) cues were incorporated in identifying non-occluded regions. This method is inspired from the observation that occluded regions often cause significant motion discontinuity in flows while occluding obstacles are closer (in depth) to the camera than occluded objects. This method requires the motion and depth information and thus is not applicable for static images.

Individual classification scores of separate parts are often not reliable in occlusion. To overcome this situation, Ouyang and Wang [114] proposed a three layer graph modelling both the appearance and visibility relationship between body parts. In the model, parts at higher layers can include parts at lower layers. The model was trained similarly to deep belief networks [149] and then used to determine occluded parts. By using this model, the classification of a body part is verified via different combinations of that part with other parts. Other methods, e.g. [53], empirically show that they can somehow deal with occlusions though there is no explicit occlusion handling mechanism in those methods.

4.2.2. Inference-based Occlusion Handling

For example, the occlusion inference was formulated as maximising a joint likelihood of human hypotheses in [32], [105], [20] and a greedy-based inference algorithm was used to obtain the optimal solution. The optimisation was performed by verifying whether each hypothesis should be added (e.g. [105], [20]) or removed (e.g. [32]) at a time to increase the joint likelihood and update the occlusion map accordingly. However, the verification of each hypothesis was performed only once. Thus the hypothesis can be rejected or accepted without consideration of the global optimal configuration of available hypotheses. In [123], human objects in inter-object occlusion were modelled in a Bayesian network and the occlusion inference was formulated as estimating a marginal probability of the image observation. Mean field algorithm was adopted to approximate the estimation. However, this method aims to reduce false alarms through the occlusion reasoning process rather than recover missed detections caused by occlusions.

In [117], [126], [119], a two-person detector was employed together with the typical single human detector to detect two-person patterns in which one is occluded by the other one. To model various occlusion situations, mixture models were used in [126], [119]. The main difference between [117], [126] and [119] is that the location of each human in the two-person detection results is also considered in both training and testing in [117], [126]. The visibility relationship between overlapping pedestrians was also exploited in [120] in which, given two overlapping pedestrians, the visibility of a part of a pedestrian helped to infer the presence of another part of the other pedestrian. Moreover, for each pedestrian, the graphical model proposed in [114] was employed to model the visibility relationship between parts in a same human object.

In [108], a grammar based model was proposed to model both the articulation and occlusion of the human body. In the method, occluders were also explicitly modelled and learned together with the body parts. An advantage of this method is that the model can be learned from weakly-labelled data, i.e. only the bounding boxes of human objects are used.

In general, the above approaches require a pre-defined set of possible occlusion types. In addition, they are not able to handle self-occlusions in which a body part of a human object is occluded by other parts of the same object. Note that self-occlusions often happen in high interaction scenes, e.g. sport videos, in which human objects appear in significantly varied and unusual poses and viewpoints.

4.3. Reduction of Computation in Detection

Computational speed is a crucial factor for real-time human detection applications such as on-board pedestrian detection. In [9], a comparison of several pedestrian detection methods on the computational efficiency was performed. The number of frames (of 640×480 pixels) processed in one second was used as the measure of the speed. This experiment was conducted on the Caltech dataset (more details of this dataset are presented in section 5.1). Figure 17 shows this comparison.

In general, the window-based detection approach is most widely adopted in human detection systems due to its simplicity, especially when only single images are given. However, this approach suffers from its computational burden due to a vast number of windows to be processed. To overcome this issue, efficient sliding window-based methods have been proposed. The main strategies include, reducing the computation required for each window and/or reducing the number of windows.

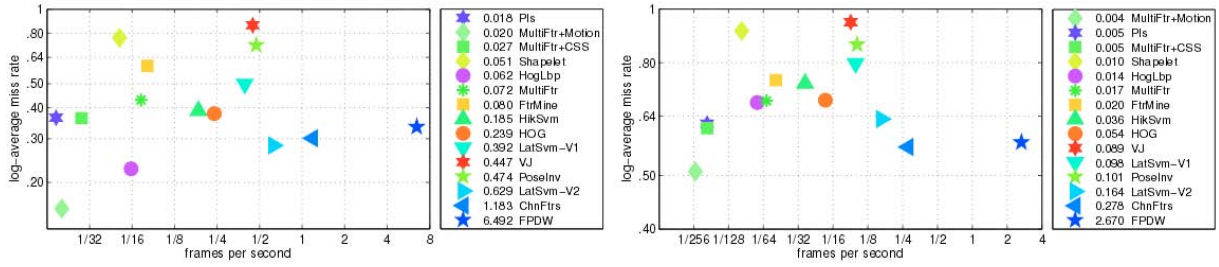


Figure 17: Computational speed of some pedestrian detection methods on the Caltech dataset. Left: for pedestrians taller than 100 pixels. Right: for pedestrians taller than 50 pixels. The computational complexity was measured on a 640×480 image. In the plots, the horizontal axis indicates the speed while the vertical axis represents the miss rate (please see section 5.1 for more details on the detection accuracy measures). The lower the miss rate is, the higher the accuracy is. Methods used in this comparison are aliased as VJ [68], HOG [15], Shapelet [48], FtrMine [54], HikSvm [94], LatSvm-V1 [95], MultiFtr [97], HogLbp [81], Pls [102], ChnFtrs [84], PoseInv [105], FPDW [106], MultiFtr+CSS [67], MultiFtr+Motion [67], LatSvm-V2 [11]. This figure is from [9] and best viewed in colour.

4.3.1. Reduction of Per Window Computation

For the first strategy, i.e. reducing the computational complexity of processing each window, several methods use different descriptors and classifiers with different levels of complexity when processing windows. The aim is to filter out obvious non-human windows as early as possible. For example, in [101], a two-stage method was proposed in which a linear SVM was first employed to obtain an initial set of candidate regions with high confidence scores. Those candidate regions were then re-classified by a more sophisticated but robust classifier (e.g. non-linear SVM). In [104], a coarse-to-fine model was proposed in which later detection results (at finer resolutions) were searched based on the best results obtained at coarser resolutions. In [116], windows were processed at three different levels. The higher the level is, the more complicated features, classifiers, and finer scale and spatial strides are used. For example, at the first level, the difference of the LBP histograms between consecutive frames was used to identify regions with sufficient motion. The dense window scanning approach with HOG, Haar, and SVM was then used at the second level. At the third level, windows classified as human at the second level were re-confirmed by using the classification scores and optical flows obtained on nearby frames.

Another approach to reducing the computation in processing windows is to optimise the calculation of features. This is especially crucial when the features are extracted at multiple scales and in a non-grid based manner. Integral images are a beneficial means to save the computational cost when the features are computed on overlapping local image regions of various sizes, e.g. [68], [91], [96], [84], [100]. To further improve the efficiency in computing features at multiple scales, features at nearby scales were approximated via extrapolation in [133]. Since the size of the human object is unknown, in the window-based detection approach, one can either process an input image at multiple scales with a fixed size detection window, e.g. [15], or fix the input image's size and vary the detection window's size, e.g. [68]. The former requires more computation while the latter degrades the quality of some features, e.g. HOG-like features due to the blurring effects. Note that theoretically both approaches should have similar performance. However, different performances are obtained in practice. In [106], [133], by approximating the features at various scales and employing integral images, the scale space was quantised at less discrete levels and thus the number of scales of the input image was significantly reduced. This idea was also applied in [115] for reducing the quantised scale space of the detection window.

4.3.2. Reduction of Window Candidates

Limiting the search space, i.e. reducing the number of windows, can be done by identifying image regions that potentially contain the human object. In [150], the input image was segmented into superpixels [151]. Those superpixels were then merged to form potential regions using constraints on the appearance (e.g. smooth boundaries, similar colour, brightness, texture), geometric, and depth information. Similarly, the size and depth of pedestrians in the scene were estimated using camera parameters and used to narrow the search space in [111]. Readers are referred to [111] for implementation details. Cascade AdaBoost [68], [91] is often used to trim off windows to be processed and thus accelerate the detection. In contrast to the traditional cascade approach, in [110], the cascade classifier could be terminated early when the human object or background could be classified with high confidence. In [115], a soft-cascade strategy was proposed in which rejection thresholds were adaptive to different stages of the cascade framework and could also be learned. As reported in [115], by efficiently computing features and using soft-cascade AdaBoost, the pedestrian detector could achieve 50 fps (frame-per-second) for monocular images and 135 fps for stereo images on a CPU + GPU enabled desktop computer while the detection accuracy was just slightly degraded. When the point-based descriptor with feature histogram is used (e.g. [96], [101], [100]), the method in [96] can be used to efficiently determine the bounds of the detection windows. In [100], in addition to estimating the bounds of human hypotheses similarly to [96], the ISM (proposed in [16]) was applied to verify the spatial consistency between the local features extracted at key points.

4.4. Use of Context Information

Context information has also been exploited to enhance the performance of object detection. Context of an object is referred to as the environment surrounding the object and in some cases can help to verify the presence of the object. For human detection, the context information can be integrated in the human descriptor or augmented as an additional cue to reasoning the existence of the human object in the scene. For example, in [113], the detection window was extended to include the background information and HOG feature was computed on both the human candidate region and extended regions. In addition, a so-called classification context feature was proposed. Specifically, the classification context of a candidate human object was defined based on the distribution of the classification scores of windows containing that human in the spatial domain and across multiple scales. Thanks to the classification of context feature proposed in [113], Chen et al. [129] constructed the context information by encoding the co-occurrence of the classification context features. In [124], the context information was computed in multiple scales and integrated in a deep architecture. The information of non-human objects can be also useful to verify the existence of nearby human objects. The work in [128] is an example of this approach. In particular, the spatial relationship between pedestrians and cars was exploited. This is because, compared with the human object, cars are easier to be detected. Given car locations, many false alarms, e.g. detections at wheels locations, can be safely filtered out.

5. Datasets, Tools, and Evaluation Schemes

5.1. Datasets

Over the last decade, a number of datasets have been made publicly available for evaluating human detection algorithms. These datasets are collected from different scenarios and thus can be used as

benchmark for a wide range of applications including image retrieval, video surveillance, and driving assistance. For example, general purpose person detection algorithms for image retrieval can be evaluated using the MIT [51], INRIA [15], Penn-Fudan [46], USC-A [32], USC-C [31], and PASCAL VOC [152] databases. For surveillance applications, there are USC-B [32] and CAVIAR [19] datasets. For pedestrian detection for driving assistance, the Caltech [9], TUD [83], CVC [8], DaimlerChrysler (DC) [145], and ETH [23] datasets are suitable.

These datasets also complement each other by presenting various complexities associated with the human form - pose, viewpoint, appearance and occlusion. In addition, they expose challenges arisen from environment such as various illumination conditions and cluttered background. For example, in the MIT and USC-A dataset, humans are shown mainly in the frontal or rear view. The INRIA, Penn-Fudan, USC-C, and PASCAL VOC dataset contains humans in various poses and viewpoints. In the USC-B, CAVIAR, PASCAL VOC, and Caltech datasets humans may be occluded. The USC-B and CAVIAR include footage videos in which humans are occluded by other humans. The Caltech dataset contains cases of pedestrians partially occluded by cars, trolleys. Compared with the USC-B, CAVIAR and Caltech dataset, occlusion cases in the PASCAL VOC happen due to the image borders, non-human objects, the interaction between humans, or even due to the human pose and camera's viewpoint. Figure 18 illustrates some typical images of the publicly available datasets.

Other factors on the datasets include the resolutions and formats (e.g. static images, videos captured by a static/moving camera, stereo images). The size of human objects varies with the camera resolution and the distance from the objects to the camera. For example, in most of driving assistance systems designed to avoid collision, pedestrians need to be detected at sufficient distance from the camera. Thus, the humans in this application are usually presented in small size, e.g. less than 80 pixel height in relative to a 640×480 pixel image. This scenario is represented in the Caltech, TUD, CVC, and DC datasets. Left and right image sequences captured by a stereo camera are provided in the ETH dataset. In the TUD dataset, a training set of pairs of images (TUD-MotionPairs) used to compute optical flows is included. Table 5.1 summarises the common publicly human object datasets.



Figure 18: Sample images of publicly available datasets. Note that for some datasets, e.g. CVC, DC, and MIT, cropped images are provided.

Table 2: Publicly available human object datasets. The column Occ is checked with Y if the dataset contains occluded humans. The occlusion types (1 indicates inter-object occlusion and 2 represents self-occlusion) are also provided.

Dataset	Format (I: image V: video S: stereo)	Occ	Spectrum (C: colour G: gray)	Training		Testing			URL
				#humans	#neg images	#humans	#neg images	#full images	
MIT [51]	I		C				924		http://cbcl.mit.edu/cbcl/software-datasets/PedestrianData.html
INRIA [15]	I		C	2416	1218	1126	453	228	http://pascal.inrialpes.fr/data/human/
CAVIAR [19] ⁴	I	Y (1)	C			5614		1590	http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/
USC-A [32]	I		G			313		205	http://iris.usc.edu/Vision-Users/OldUsers/bowu/DatasetWebpage/dataset.html
USC-B [32]	I	Y (1)	G			271		54	as USC-A
USC-C [31]	I		G			232		100	as USC-A
Penn-Fudan [46]	I		C			345		170	http://www.cis.upenn.edu/~jshi/ped_html/
DC [145]	I		G	14400	1200 + 15000 (samples)	9600	10000 (samples)		http://www.lookingatpeople.com/download-daimler-ped-class-benchmark/
TUD [83] ⁵	V		C	3552 × 2	192 × 2 + 26 × 2	1326		508 × 2	http://www.d2.mpi-inf.mpg.de/tud-brussels
Caltech [9]	V	Y (1)	C	192000	61000	155000	56000	65000	http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/datasets/USA/
CVC [8]	I		C			2000	6175 (samples)		http://www.cvc.uab.es/adas/
ETH [23] ⁶	S		C			12000		1804	http://www.vision.ee.ethz.ch/~aess/dataset/
PASCAL [152] VOC ⁷	I	Y (1,2)	C	4194	1994	4372		2093	http://pascal1in.ecs.soton.ac.uk/challenges/VOC/

⁴This set contains a lot of video sequences. Here, only one of these sequences, the *OneStopMoveEnter1cor* sequence (in corridor view), which has been used commonly in recent human detection methods, e.g. [19], [20] is presented.

⁵Pairs of positive samples and images are given. Thus, the real numbers of those samples and images are doubled (i.e. ×2)

⁶The numbers of humans and images of this dataset are summarised and reported in [9].

⁷This dataset was first released in 2005 and annually updated until 2012. The data reported here is from the most recent version in 2012. Since 2008, the annotations of the test sets have not been released publicly. Instead, the detection results are to be submitted to the PASCAL VOC evaluation server at <http://host.robots.ox.ac.uk:8080/> for performance evaluation.

Included in the datasets, the ground-truth results are annotated and available for evaluation. Annotated human objects are often represented by bounding boxes and stored in different formats. For example, the XML file format which makes the annotations portable and easy to extend (e.g. to describe other properties of the humans such as occlusion, group of people, etc.) is used in the CAVIAR, USC-A, -B, and -C datasets. In the INRIA, TUD, ETH, Penn-Fudan, and PASCAL VOC datasets, the annotation is written in the PASCAL format [152]. Figure 19 shows both XML and PASCAL annotation formats.

Although the main aim of evaluating human detection methods is to measure the accuracy in locating the entire human bodies, in the PASCAL VOC dataset, bounding boxes corresponding to body parts, called person layout, are also provided. In addition to the bounding boxes, in some datasets, e.g. the Penn-Fudan and PASCAL VOC, contours and segmentation masks are also available (see Figure 18). The detection results (bounding boxes) of many pedestrian detection methods and Matlab performance evaluation code on several datasets such as the INRIA, DC, TUD, Caltech, ETH can also be found at http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/.

5.2. Tools and Evaluation Schemes

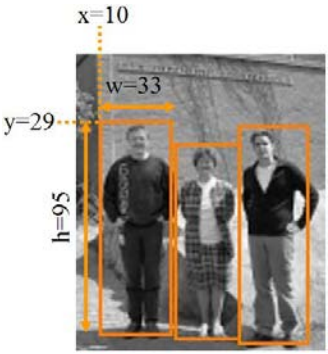
Evaluation of a human detection algorithm requires to annotate (label) human objects in the images/videos in chosen datasets. One of the commonly used tools for image annotation is the LabelMe [153], which is a web-based annotation tool providing functionalities to draw and edit objects' bounding boxes, query images, and browse the database. LabelMe also supports semi-automatic labelling in which some available detection algorithms can be used to generate the initial detection results and the results can then be validated and edited by users.

To quantify the detection performance of human detection algorithms, a number of measures have been proposed. Since the problem of human detection can be formulated as binary classification in which candidate regions are either human or non-human, classification/recognition accuracy can be considered as an indicator of the detection performance. In some datasets (e.g. INRIA, CVC, and DC), where cropped humans (positive samples) and non-humans (negative samples) are given, the evaluation can be performed simply as testing an object recognition system. In this case, the detection performance can be represented by the receiver operator characteristic (ROC) which represents the trade-off between true positive and false positive rates. Specifically, the detection rate or true positive rate (TPR) and false positive rate (FPR) are computed as,

$$\begin{aligned} \text{TPR} &= \frac{\# \text{True Recognitions}}{\# \text{Positive Samples}} \\ \text{FPR} &= \frac{\# \text{False Recognitions}}{\# \text{Negative Samples}} \end{aligned} \quad (1)$$

Note that in some methods (e.g. [15]), the true positive rate is replaced by the miss rate (MR) defined as $\text{MR} = 1.0 - \text{TPR}$. The FPR can also be considered as the false positive per window (FPPW) rate since each sample is represented by a detection window.

Although the ROC with TPR/MR and FPR/FPPW is often employed to evaluate object detection algorithms, it has drawbacks when used for comparison of detection methods on datasets which do not explicitly provide samples. As argued by Agarwal and Roth [154], the number of negative samples generated by scanning an input image using a detection window depends on many factors such as the size of the detection window, the number of scales, scale stride, and spatial stride. Those parameters can be



```
<?xml version="1.0"?>
<ObjectList>
  <Object>
    <Rect x="10" y="29" width="33" height="95"/>
  </Object>
  <Object>
    <Rect x="43" y="39" width="30" height="87"/>
  </Object>
  <Object>
    <Rect x="71" y="31" width="32" height="97"/>
  </Object>
</ObjectList>
```

(a)



```
# PASCAL Annotation Version 1.00

Image filename : "Test/pos/crop_000002.png"
Image size (X x Y x C) : 333 x 531 x 3
Database : "The INRIA Rhône-Alpes Annotated Person Database"
Objects with ground truth : 1 { "PASperson" }

# Note that there might be other objects in the image
# for which ground truth data has not been provided.

# Top left pixel co-ordinates : (0, 0)

# Details for object 1 ("PASperson")
# Center point -- not available in other PASCAL databases -- refers
# to person head center
Original label for object 1 "PASperson" : "UprightPerson"
Center point on object 1 "PASperson" (X, Y) : (176, 112)
Bounding box for object 1 "PASperson" (Xmin, Ymin) - (Xmax, Ymax) :
(110, 80) - (214, 469)
```

(b)

Figure 19: An image with annotated humans from the USC-A and INRIA dataset written in XML format (a) and PASCAL format (b).

setup differently among different detection systems. Thus, when the negative samples are not available, the precision-recall (PR) should be used as a metric to evaluate and compare detection algorithms. A definition of recall and precision is given as follows,

$$\begin{aligned}\text{Recall} &= \frac{\#\text{True Positives}}{\#\text{Annotated Humans}} \\ \text{Precision} &= \frac{\#\text{True Positives}}{\#\text{True Positives} + \#\text{False Positives}}\end{aligned}\quad (2)$$

A detection result (e.g. a bounding box) is considered as true positive if it matches a ground-truth and false positive (or false alarm) otherwise. As can be seen, the recall is the TPR while the precision does not depend on how many detection windows are sampled from the test image.

In [152], [9], a detected bounding box (BB_{dt}) is considered as matched with an annotated bounding box (BB_{gt}) in the ground-truth if the following inequality is satisfied,

$$\frac{\text{area}(BB_{gt} \cap BB_{dt})}{\text{area}(BB_{gt} \cup BB_{dt})} > \lambda \quad (3)$$

where λ is a user-defined value which represents the overlapping (tightness) of the detected bounding box on the true result. This value is often set to 0.5 in evaluation and comparison of object detectors [9].

Similarly to the PR measure, Dollár et al. [9] proposed the use of false positive per image (FPPI) instead of FPPW vs. the MR to evaluate the overall detection performance. In FPPI evaluation, nearby detection windows which are classified as humans are merged using a postprocessing step. Non-maximal suppression (NMS) is the most widely used postprocessing algorithm. Basically, NMS is conducted using mean-shift or pairwise max suppression [9]. The detection performance is then evaluated based on the merged detection results. It is shown in [9] that there is a discrepancy between FPPW and FPPI evaluation, i.e. if a method is better than another one when FPPW is used, it may not outperform the other method when FPPI is used. Figure 20(a) illustrates the discrepancy in the comparison of several human detection methods (e.g. “Shapelet” and “HKSvm” methods) on the INRIA dataset using FPPW and FPPI. Moreover, in [9], Dollár et al. argued that detection windows generated by FPPW evaluation were not the same as those generated during FPPI evaluation and that FPPW evaluation did not cover all the cases considered by FPPI evaluation. Figure 20(b) illustrates the cases that are often found in FPPI evaluation but not included in FPPW evaluation. For example, humans in the positive samples in FPPW evaluation are supposed to be well centred in the detection windows. Hence, false positives caused by scaling and translating the humans and by detecting subparts will not happen.

However, we have found that the above situation (i.e. the incoincidence of detection windows generated in FPPW and FPPI evaluation process) happens due to the way of constructing the positive and negative samples. Suppose that we are given an image with annotated objects (i.e. true objects), we scan this image using a detection window at various scales and locations. Each window is then labelled as positive or negative window based some criterion, e.g. the one presented in (3). This process will result in the sets of positive and negative samples (windows). Note that, based on the criterion in (3), the ratio of a positive window and its contained human object may vary and the human object may not be centred in the window either. In addition, negative windows not only represent the background but also capture parts of human objects. We illustrate this process in Figure 21. By using this methodology, the positive and negative sets would include all cases mentioned in Figure 20(b), i.e. the combination of the positive and negative sets would be same as the set of windows generated by FPPI evaluation. Therefore, we

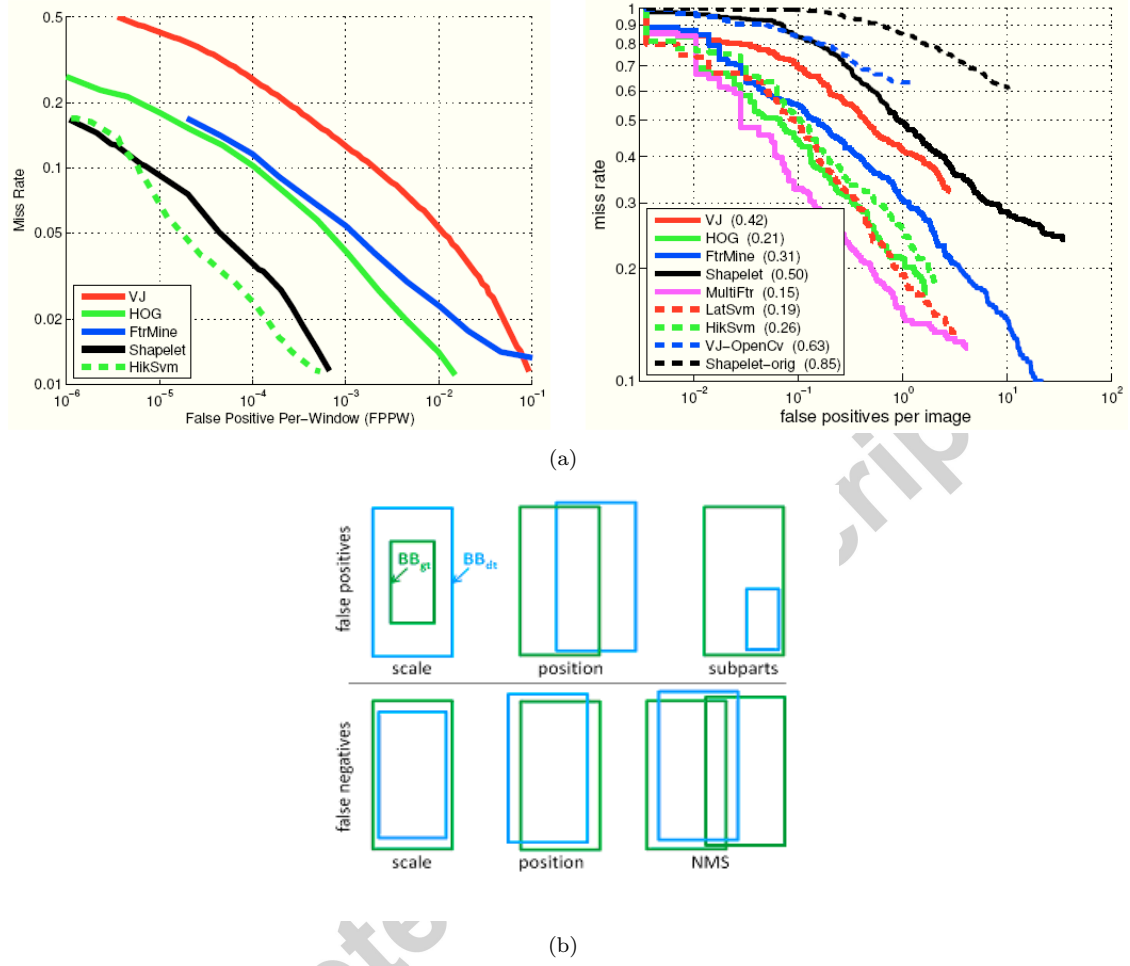


Figure 20: (a) FPPW (left) and FPPI (right) results on the INRIA dataset (from [9]). (b) Untested cases in FPPW evaluation (from [9]); BB_{dt} and BB_{gt} denote the detected bounding box and ground-truth bounding box respectively.

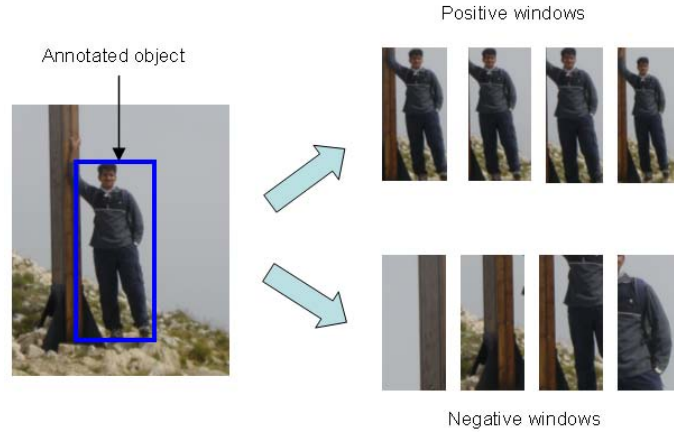


Figure 21: Illustration of creating the positive and negative windows from annotated objects.

believe that FPPW evaluation is not flawed in detection performance evaluation. Rather, it just stands from a different point of view, i.e. the classification viewpoint. Moreover, in some applications, e.g. image retrieval, evaluation using FPPW is reasonable and probably preferred than FPPI.

FPPW evaluation also has advantages compared with FPPI evaluation. Firstly, FPPW can be used to compare features, human descriptors, and classifiers. Indeed, if all detection algorithms are implemented using the same experimental setup (i.e. same number of scales, same spatial and scale stride), then the same positive and negative samples will be created and FPPW is valid for evaluation and comparison. Secondly, FPPW evaluation is not affected by the NMS while the discrepancy between FPPW and FPPI may be due to the NMS. We illustrate the effects of the NMS (for both cases: same NMS and different NMS) on the detection performance in Figure 22. Figure 22(a) represents a case in which the same NMS is used. In this figure blue rectangles (in dot pattern) represent the annotated object, green and red rectangles are the true and false detection results respectively. The criterion used to determine true and false detections is presented in (3). Figure 22(a) explains why there is the discrepancy between FPPW and FPPI evaluation when ranking detection methods. As presented in Figure 22(a), algorithm 2 has a better FPPW indicator whereas algorithm 1 has a better FPPI indicator. Figure 22(b) illustrates the effect of different NMS methods on the detection results. As shown, these two NMS methods could produce different detection results.

To conclude, for an accurate and fair comparison, human detection methods should be evaluated using both FPPW and FPPI measure. In addition, for FPPW evaluation, positive and negative windows should be generated so that they are same as windows obtained by FPPI evaluation process.

6. Trends, Issues, and Future Work

6.1. Features and Descriptors

Development of human detection over the past decade is mainly driven by two seminal works. First, for the features, HOG [15] is mostly used although other feature types, e.g. LBP and Haar-like, have also shown their potential in many cases. This is probably because the success of the HOG feature has been confirmed on different datasets. It is obvious that the use of appropriate features is a key to the discriminative power of human descriptors. However, this often requires a manual design of the features and empirical selection and validation. Recently, automatic selection of features for object description based on sparse coding technique has been studied and promising results were gained [155], [156].

Grid-based descriptors are often used in human detection methods. However, as presented, grid-based descriptors are sensitive to heavy deformations of the human object and variations of the viewpoint. Point-based descriptors would be more adaptive to those deformations and variations. However, existing key point detectors, e.g. [141], [139], [140], extract key points independently by using only low-level information, e.g. gradient, image intensity. Therefore, the detected key points may not be part of the human object. Development of object knowledge-based key point detectors using high-level information of the object of interest, e.g. shape, texture, etc., thus deserves in-depth studies.

The second important work that has made an influence in human detection studies is the deformable part model (DPM) proposed in [95]. The DPM has currently received considerable attention due to its robustness in dealing with articulation. However, as shown in [85], for describing pedestrians at low resolutions and less deformation, a holistic grid-based descriptor with careful designs of feature normalisation and classifiers could achieve high detection accuracy and less computation in comparison to the DPM.

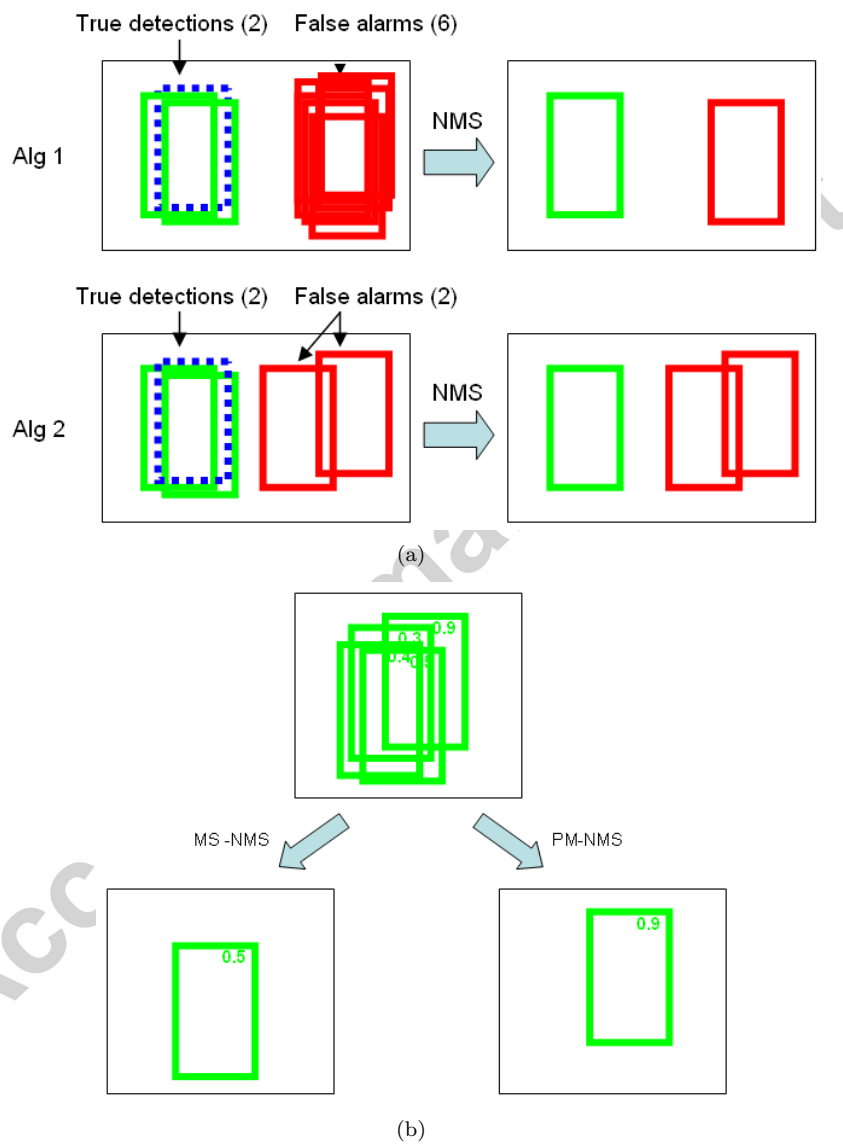


Figure 22: Effects of the NMS. (a) Discrepancy between FPPW and FPPI performance even when the same NMS is used. (b) Different NMS methods: mean-shift (MS) vs. pairwise max (PM) suppression. The number on each detection window indicates the detection score. The higher the number is, the higher the probability the window encloses a human object.



Figure 23: Some challenging examples of humans in different views (the first row), poses, and occlusions.

It is worthwhile to notice that the robustness of the features and human descriptors has often been confirmed via environmental challenges such as illumination conditions, cluttered background in existing work, while image/video resolution has been underestimated. In fact, image/video resolution also makes an impact on the performance of human detection. Specifically, for window-based detection methods, e.g. [15], the human object is assumed to fit well the detection window. Thus when the input image/video is captured at low resolutions, the image/video needs to be scaled up and the features (especially edge-based features such as the HOG [15]) are deformed due to the upsampling or interpolating. It is shown in [9] that the performance of pedestrian detection methods is degraded correlatively to the reduction of the pedestrian's sizes.

Deep structures, e.g. convolutional neural networks, have shown a research trend in multi-class object recognition and detection [157], [158], [159] and achieved remarkable performance on challenging datasets, e.g. the PASCAL VOC dataset. The benefits of deep structures are known as their capability of self-constructing the object descriptor and learning higher-level features which are not explicitly and directly provided. Some successes of this approach have been found in the recent pedestrian detection methods such as [124], [13], [134], [137] in which the deep models were adapted to fit the pedestrian object. However, the architecture of the deep models used in those methods needs a hand-crafted design, e.g. the number of parts and modes of combination of parts are fixed and determined in advance. In addition, training deep and complicated structures may be intractable.

6.2. Pose and Viewpoint Variation

It is clearly to see that most of the current approaches are rather appropriate for pedestrian detection in which the human object is implicitly supposed to be in the upright-standing pose. However, this assumption is not held in other applications, e.g. image retrieval, video indexing, and human action analysis in sports. Figure 23 shows some examples that do not happen in pedestrian detection but may be common for other applications. As shown in the literature, only few methods, e.g. [95], [11], [127] (with DPM) and [99], [12] (with poselets), show their capability in somewhat dealing with such challenges. However, heavy deformations and variations as presented in Figure 23 remain challenging and unsolved. Specifically, there exist issues that are worthy of being considered.

- Part-based models are robust and adaptive to the deformations of non-rigid objects and thus desirable for human descriptors. However, these models often require that the number of parts and part types (e.g. head, torso, legs) must be determined in advance. In fact, the number of parts varies to different poses and viewpoints. For example, one arm of may not be visible when the human object is in a profile view and the face is not observable in a back-facing pose. Moreover, a part can also appear differently in different poses and viewpoints. For example, the legs in a running and standing pose have different appearance. This challenge can be found easily in complicated repertoire of athletes in sport videos.
- To overcome the above issue, using multiple part descriptors and classifiers to describe and classify each part in multiple poses and viewpoints could be considered, e.g. [99], [12]. However, this approach requires more annotated data of the parts in different poses and viewpoints. Meanwhile, available datasets only provide the class label of training samples, i.e. human/non-human samples or annotated bounding boxes containing humans. Furthermore, manually collecting and annotating parts is labour expensive. An automatic labelling system for object parts annotation in various poses and viewpoints is thus worthy to be developed. Synthetic 3D human models would be useful for

such a system in which the models can generate images of a part in different poses and viewpoints simply by varying the model parameters.

- Another issue with part-based models is how to model the human body using parts. Graphical models, e.g. [142], [47], [11], [127], are often used for this purpose and the task of human detection is formulated as the inference in the models. Graphical models embody constraints on the spatial layout of parts in a given pose and the appearance consistency of parts in a given viewpoint. However, what model (pairwise-, star-model, etc.) should be the best to represent the constraints on the configuration of parts in articulation and affordable for the computations in the model? In addition, is there a mechanism for automatic selection of such a model? Dense and complicated models better capture the above constraints but incur more computational complexity for the training and inference task.
- Existing part-based approaches construct the final detection results based on the part detections and thus their performance depends on part detectors/classifiers. However, part detectors/classifiers may not work perfectly and hence may produce spurious results while correct parts may be missed. Existing part-based approaches can compensate the mistake made by one part detector by considering the results of other part detectors. However, they are not able to recover missing parts. Note that the performance of detecting parts is also an important indicator to evaluate part-based models.
- Detecting humans in various poses is extremely difficult in high interaction scenes (as shown in Figure 23) where the body parts of different humans are spatially nearby and part-based models may group parts of different subjects into one body structure. This difficulty has not been considered in the current approaches.

6.3. Occlusion

Detecting occluded humans is one of the most difficult problems and appears unsolved though a few attempts have been proposed to address this problem. There are two issues related to existing occlusion handling methods.

- The first issue is how to model occlusions. As shown in the literature, part-based human descriptors are adopted to describe humans under occlusion. However, it is possible to craft an application-dependent part-based model, it is hard to define a robust model that is applicable for a wide range of applications and to cover various occlusion cases. For example, in [32], [105], a full human body was decomposed into three parts: upper part (head-shoulder), middle part (torso), and lower part (legs). In [123], a human object included upper and lower part, left and right part. These models are suitable for footage videos in surveillance systems as shown in Figure 2(b) in which humans are often captured at far distance from the camera, and thus details, e.g. human face and arms, are hard to be identified. The model proposed in [117], [126] describes cases in which a pedestrian is captured in a profile view and occluded by another pedestrian passing him/her in the same profile view. This occlusion pattern is often seen in pedestrian detection for driving assistance systems. In all, the above models only capture occlusion cases which can be due to the interaction between humans. However, they are not suitable to model intra-object occlusion (i.e. self-occlusion) and even not able to handle severe cases, e.g. only the human face or part of a limb is visible (as

illustrated in the 3rd image in the 1st row and the 3rd image in the 3rd row in Figure 23). We notice that this type of challenge is often found in image/video retrieval applications.

- As important as selection of a part-based model covering possible occlusion cases, determining occlusion appears a key issue in occlusion handling. Existing methods identify occlusion by using the classification scores of parts, e.g. [81], [73]. As a consequence, the performance of occlusion handling strongly depends on the reliability of part classifiers. It is also important to note that the pose and viewpoint information are useful to occlusion reasoning. For example, the upper-legs of a human object cannot be observed when the object is in a sitting pose and the torso in a profile view indicates that one of the arms is likely occluded.

6.4. Fine-grained Human Detection

Existing human detection methods only produce bounding boxes as the detection results. However, human detection is a crucial component in many applications, and hence, the detailed information derived from the detection such as the human face, segmentation mask, viewpoint and pose information, visible parts and 3D location (if available) of parts, etc., would be worth for the applications. Such information is referred to as fine-grained information. For example, in surveillance systems using multiple cameras, the face's angle of a detected human captured by a camera can be used to activate another proper camera for detecting the person identity. 3D locations of parts can be used in interactive games and are important for pose estimation. Several efforts of this stream can be found in the literature though the current achievements are limited in the richness of the retrieved information and the complexity of applications. For example, Enzweiler and Gavrila [107] proposed a pedestrian body orientation estimation method integrated in human detection. The body orientation of a pedestrian was estimated using a Gaussian mixture model of body orientation constructed on the training samples. The proposed method targets on only four orientations: front, left, back, and right views. Part-based models, e.g. pictorial structure model [142] and DPM [95], were also applied for pose estimation in [47] and [127] respectively.

It is also clearly to see the contribution of fine-grained information to enhance the performance of human detection. For example, the pose information can be used to validate the existence of a human object and the viewpoint information with 3D location of parts is useful for occlusion reasoning, especially for self-occlusion. However, pose estimation is often considered a subsequent task of human detection in the current approaches and thus pose and viewpoint information is not effectively exploited. In recent years, research on understanding humans, e.g. human pose estimation for Kinect, medical image analysis, has been strengthened by advances in 3D technologies. In addition, new inexpensive cameras such as Microsoft Kinect and ASUS Xtion allow users to acquire the 3D data of a scene much more easily than ever. These developments have made fundamentals and thus would enable further and in-depth research on fine-grained human detection.

7. Selection of Algorithms for Applications - A Brief Guideline

Since various human detection methods focus on different aspects, to maximise the effectiveness of the detection task in a particular application, existing methods should be chosen carefully based on the characteristics and purposes of the application. The factors to be considered in selecting a human detection method for an application includes the variation of the human poses and viewpoints, occlusion, size of the human object, full or partial detection (e.g. only the head-shoulder is of interest), real-time

requirement, and format of the input (i.e. static image, video, stereo, etc.). We also provide the URL of the available source/binary code of some human detection methods in Appendix A.

For the applications (e.g. interactive games) in which only some parts (e.g. the upper body) of a human object are to be detected and the humans appear in large size relatively to the input image/video frame, template matching-based methods, e.g. [27], are preferred. This is because template matching-based detectors do not require off-line training, can be adaptive on-the-fly to user-provided templates and work reliably (due to less background). The recent template matching method proposed in [160] has shown its potential in object detection with high speed detection and low miss rate.

For the applications in which the humans appear in high variance of poses, methods such as [11], [12], are more suitable. Compared with the DPM [11], the method in [12] can be more easily adapted to various poses by adding new poselet types (or removing current ones) without retraining existing poselet detectors. Experimental results on the PASCAL VOC datasets have also shown that the poselet-based human detector in [12] outperformed the DPM. However, both the methods are not real-time. Especially, the poselet detectors in [12] run independently, thus the computational complexity of the detection task strongly depends on the number of poselet detectors (i.e. poselet types) used. We have run the provided code of the method [12] (see Appendix A for the source code) on an Intel(R) Core(TM) i7 2.10GHz CPU computer with 8.00 GB memory. Experimental results on the PASCAL VOC 2007 dataset have shown that, on the average, with 150 poselet detectors, 15 seconds were spent for detecting poselets on each image. Since the poselet detectors act independently, the method could be sped up by optimising the code on computers enabling parallel computation.

For those applications that have less view and pose variation, e.g. pedestrian detection in driving assistance systems, simpler methods such as [84], [85], [14] are more appropriate. This is because, in pedestrian detection, human objects are often present in upright-standing pose, in common viewpoints (e.g. frontal, back, left, right view), and need to be detected at far distance. However, when the humans are very far from the camera, the poses and viewpoints cannot be clearly observed, and therefore, complicated methods such the DPM [11] may not show much advantage. When the input of the system is video sequences captured by a moving camera, the pedestrian detection method [72] using motion features can be considered.

For the applications that are sensitive to response time, e.g. assisted automotive applications or interactive games, computational speed is the most crucial factor. Recent methods such as [104], [115], [133] show promising results on these applications. For example, Benenson et al. [115] show that their method could achieve 50 fps (frame-per-second) for monocular images and 135 fps for stereo images on a CPU + GPU enabled computer with reasonable detection accuracy. The method in [14] is also simple for implementation, effective for detection, and efficient for computation and thus potential for assisted automotive systems. This method was implemented in Matlab and hence could be accelerated in C++ on GPU enabled computers. Readers are referred to Figure 17 for the comparison on the processing time of pedestrian detectors on the Caltech dataset provided in [9].

In some applications, e.g. surveillance application, there may be a need for locating the body parts in addition to the whole body for estimation of the human pose and/or recognition of the human action. Methods such as [47], [12], [127], [135] should be considered for those applications. In [135], the keypoints including the joints on the body, the eyes and nose, are also located. As shown in [127], [135], the head and torso are detected more accurately compared with other parts. Therefore, face detection [70] could also be employed to enhance the accuracy of detecting parts; locations of parts can be inferred based on

location of the human face. Note that, some methods, e.g. [127] may be sensitive to parameter settings and thus a careful design of the training dataset, classifier, etc. need to be taken account. Validation sets with images/videos similar to the application domain should also be used to justify models and parameters.

8. Conclusion

The problem of human detection has received considerable attention in the last decade due to its wide range of applications. This paper presents a comprehensive review of state-of-the-art methods with a focus on human descriptors. It provides an insight into the success as well as failure of the existing methods in different application scenarios. We also consider recent advances in classifiers and learning algorithms, occlusion, real-time human detection, and use of context information. Research trends, issues, and future research directions are highlighted. Furthermore, a brief guideline is provided for the selection of appropriate methods based on the characteristics of the target applications. In all, the development of human detection has come to a stage where effective detectors can be constructed for applications where full upright body is visible or upper body (with less deformation) is to be detected.

Appendix A. Code/Software

Below is the list of publicly available binary/source code (with the URL) of some human detectors.

Table A.3: Software and source code of some human detection methods.

Method	URL	Code type/ Programming language
[15] ⁸	http://pascal.inrialpes.fr/soft/olt/	Binary, C++
[16]	http://www.vision.rwth-aachen.de/software/ism	Binary, C++
[81]	http://web.missouri.edu/~hantx/	Binary
[47]	http://www.d2.mpi-inf.mpg.de/andriluka_cvpr09	C++
[11]	http://www.cs.berkeley.edu/~rbg/latent/	Matlab, C++
[103]	http://graphics.cs.msu.ru/en/science/research/machinelearning/hough	C++
[12]	http://www.lubomir.org/poselets/	Matlab, C++
[115]	https://bitbucket.org/rodrigob/doppia	C++
[127]	http://www.ics.uci.edu/~dramanan/software/pose/	Matlab
[119]	http://www.ee.cuhk.edu.hk/~wlouyang/projects/ouyangWcvpr13MultiPed/index.html	Matlab
[13]	http://www.ee.cuhk.edu.hk/~wlouyang/projects/ouyangWiccv13Joint/index.html	Matlab
[156]	http://www.ics.uci.edu/~dramanan/software/sparse/	Matlab, C
[106] ⁹	http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html	Matlab
[135]	http://www.cs.berkeley.edu/~gkioxari/	Matlab

⁸Implementation of this method has been added in the OpenCV library

⁹See the **channels** and **detector** directory

References

- [1] J. K. Aggarwal, Q. Cai, Human motion analysis: A review, *Computer Vision and Image Understanding* 73 (3) (1999) 428–440.
- [2] D. M. Gavrilu, The visual analysis of human movement: a survey, *Computer Vision and Image Understanding* 73 (1) (1999) 82–98.
- [3] L. Wang, W. Hu, T. Tan, Recent developments in human motion analysis, *Pattern Recognition* 36 (2003) 585–601.
- [4] T. B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, *Computer Vision and Image Understanding* 81 (2001) 231–268.
- [5] T. B. Moeslund, A. Hilton, V. Krger, A survey of advances in vision-based human motion capture and analysis, *Computer Vision and Image Understanding* 104 (2006) 90–126.
- [6] T. Gandhi, M. M. Trivedi, Pedestrian protection systems: Issues, survey, and challenges, *IEEE Transactions on Intelligent Transportation Systems* 8 (3) (2007) 413–430.
- [7] M. Enzweiler, D. M. Gavrilu, Monocular pedestrian detection: Survey and experiments, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (12) (2009) 2179–2195.
- [8] D. Gerónimo, A. M. López, A. D. Sappa, T. Graf, Survey of pedestrian detection for advanced driver assistance systems, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32 (7) (2010) 1239–1258.
- [9] P. Dollár, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: An evaluation of the state of the art, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (4) (2012) 743–761.
- [10] R. Benenson, M. Omran, J. Hosang, B. Schiele, Ten years of pedestrian detection, what have we learned?, in: *Proc Workshop on Computer Vision for Road Scene Understanding and Autonomous Driving*, 2014.
- [11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part based models, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32 (9) (2010) 1627–1645.
- [12] L. Bourdev, S. Maji, T. Brox, J. Malik, Detecting people using mutually consistent poselet activations, in: *Proc European Conference on Computer Vision*, 2010, pp. 168–181.
- [13] W. Ouyang, X. Wang, Joint deep learning for pedestrian detection, in: *Proc International Conference on Computer Vision*, 2013, pp. 2056–2063.
- [14] S. Zhang, C. Bauckhage, A. B. Cremers, Informed haar-like features improve pedestrian detection, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2014, pp. 947–954.
- [15] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, Vol. 1, 2005, pp. 886–893.

- [16] B. Leibe, E. Seemann, B. Schiele, Pedestrian detection in crowded scenes, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 1, 2005, pp. 878–885.
- [17] N. Dalal, Finding people in images and videos, Ph.D. thesis, INRIA Grenoble (2006).
- [18] C. Stauffer, W. E. L. Grimson, Learning patterns of activity using real time tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 747–767.
- [19] Z. Lin, L. S. Davis, D. Doermann, D. DeMenthon, Hierarchical part-template matching for human detection and segmentation, in: Proc International Conference on Computer Vision, 2007, pp. 1–8.
- [20] C. Beleznai, H. Bischof, Fast human detection in crowded scenes by contour integration and local shape estimation, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 2246–2253.
- [21] L. Zhao, C. E. Thorpe, Stereo- and neural network-based pedestrian detection, IEEE Transactions on Intelligent Transportation Systems 1 (3) (2000) 148–154.
- [22] D. M. Gavrila, S. Munder, Multi-cue pedestrian detection and tracking from a moving vehicle, International Journal of Computer Vision 37 (1) (2007) 41–59.
- [23] A. Ess, B. Leibe, L. V. Gool, Depth and appearance for mobile scene analysis, in: Proc International Conference on Computer Vision, 2007, pp. 1–8.
- [24] D. Gerónimo, A. D. Sappa, D. Ponsa, A. M. López, 2D3D-based on-board pedestrian detection system, Computer Vision and Image Understanding 114 (2010) 583–595.
- [25] L. L. W. Renninger, Parts, objects and scenes: Psychophysics and computational models, Ph.D. thesis, UC Berkeley (2003).
- [26] J. D. Winter, J. Wagemans, Contour-based object identification and segmentation: stimuli, norms and data, and software tools, Behaviour Research Methods, Instruments, and Computers 36 (4) (2004) 604–624.
- [27] D. M. Gavrila, V. Philomin, Real-time object detection for smart vehicles, in: Proc IEEE International on Computer Vision, Vol. 1, 1999, pp. 87–93.
- [28] P. F. Felzenszwalb, D. P. Huttenlocher, Efficient matching of pictorial structures, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 2, 2000, pp. 66–73.
- [29] S. Ioffe, D. A. Forsyth, Probabilistic methods for finding people, International Journal of Computer Vision 43 (1) (2001) 45–68.
- [30] D. M. Gavrila, A Bayesian, exemplar-based approach to hierarchical shape matching, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (8) (2007) 1–14.
- [31] B. Wu, R. Nevatia, Cluster boosted tree classifier for multi-view, multi-pose object detection, in: Proc International Conference on Computer Vision, 2007, pp. 1–8.
- [32] B. Wu, R. Nevatia, Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors, International Journal of Computer Vision 75 (2) (2007) 247–266.

- [33] P. F. Felzenszwalb, D. P. Huttenlocher, Distance transforms of sampled functions, Tech. rep., Cornell Computing and Information Science, <http://www.cs.cornell.edu/~dph/papers/dt.pdf> (2004).
- [34] E. Seemann, B. Leibe, K. Mikolajczyk, B. Schiele, An evaluation of local shape-based features for pedestrian detection, in: Proc British Machine Vision Conference, 2005, pp. 1–10.
- [35] J. F. Canny, A computational approach to edge detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (6) (1986) 679–698.
- [36] K. Mikolajczyk, C. Schmid, A. Zisserman, Human detection based on a probabilistic assembly of robust part detectors, in: Proc European Conference on Computer Vision, 2004, pp. 69–82.
- [37] A. Shashua, Y. Gdalyahu, G. Hayun, Pedestrian detection for driving assistance systems: Single-frame classification and system level performance, in: Proc IEEE International Intelligent Vehicles Symposium, 2004, pp. 1–6.
- [38] C. R. Wang, J. J. Lien, Adaboost learning for human detection based on histograms of oriented gradients, in: Proc Asian Conference on Computer Vision, 2007, pp. 885–895.
- [39] C. Hou, H. Ai, S. Lao, Multiview pedestrian detection based on vector boosting, in: Proc Asian Conference on Computer Vision, Vol. 1, 2007, pp. 210–219.
- [40] Q. Ye, J. Jiao, B. Zhang, Fast pedestrian detection with multi-scale orientation features and two-stage classifiers, in: Proc IEEE International Conference on Image Processing, 2010, pp. 881–884.
- [41] D. Park, D. Ramanan, C. Fowlkes, Multiresolution models for object detection, in: Proc European Conference on Computer Vision, 2010, pp. 241–254.
- [42] C. Conde, D. Moctezuma, I. M. D. Diego, E. Cabello, HoGG: Gabor and HoG-based human detection for surveillance in non-controlled environments, Neurocomputing 100 (2013) 19–30.
- [43] A. Bar-Hillel, D. Levi, E. Krupka, C. Goldberg, Part-based feature synthesis for human detection, in: Proc European Conference on Computer Vision, 2010, pp. 127–142.
- [44] A. Satpathy, X. Jiang, H. L. Eng, Human detection by quadratic classification on subspace of extended histogram of gradients, IEEE Transactions on Image Processing 23 (1) (2014) 287–297.
- [45] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (4) (2002) 509–522.
- [46] L. Wang, J. Shi, G. Song, I. Shen, Object detection combining recognition and segmentation, in: Proc Asian Conference on Computer Vision, 2007, pp. 189–199.
- [47] M. Andriluka, S. Roth, B. Schiele, Pictorial structures revisited: People detection and articulated pose estimation, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 1014–1021.
- [48] P. Sabzmejdani, G. Mori, Detecting pedestrians by learning shapelet features, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
- [49] O. Tuzel, F. Porikli, P. Meer, Pedestrian detection via classification on riemannian manifolds, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (10) (2008) 1713–1727.

- [50] D. Tosato, M. Farenzena, M. Cristani, V. Murino, Part-based human detection on riemannian manifolds, in: Proc IEEE International Conference on Image Processing, 2010, pp. 3469–3472.
- [51] C. Papageorgiou, T. Poggio, A trainable system for object detection, International Journal of Computer Vision 38 (1) (2000) 15–33.
- [52] A. Mohan, C. Papageorgiou, T. Poggio, Example-based object detection in images by components, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (4) (2001) 349–361.
- [53] P. Dollár, B. Babenko, S. Belongie, P. Perona, Z. Tu, Multiple component learning for object detection, in: Proc European Conference on Computer Vision, Vol. 2, 2008, pp. 211–224.
- [54] P. Dollár, Z. Tu, H. Tao, S. Belongie, Feature mining for image classification, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
- [55] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on featured distributions, Pattern Recognition 29 (1) (1996) 51–59.
- [56] Y. Mu, S. Yan, Y. Liu, T. Huang, B. Zhou, Discriminative local binary patterns for human detection in personal album, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
- [57] S. Hussain, B. Triggs, Feature sets and dimensionality reduction for visual object detection, in: Proc British Machine Vision Conference, 2010, pp. 1–10.
- [58] D. T. Nguyen, Z. Zong, W. Li, P. Ogunbona, Object detection using non-redundant local binary patterns, in: Proc IEEE International Conference on Image Processing, 2010, pp. 4609–4612.
- [59] A. Satpathy, X. Jiang, H. L. Eng, Human detection using discriminative and robust local binary pattern, in: Proc IEEE International Conference on Acoustics, Speech, and Signal Processing, 2013, pp. 2376–2380.
- [60] M. Heikkilä, M. Pietikäinen, C. Schmid, Description of interest regions with local binary patterns, Pattern Recognition 42 (3) (2009) 425–436.
- [61] Y. Zheng, C. Shen, X. Huang, Pedestrian detection using center-symmetric local binary patterns, in: Proc IEEE International Conference on Image Processing, 2010, pp. 3497–3500.
- [62] Y. Zheng, C. Shen, R. Hartley, X. Huang, Pyramid center-symmetric local binary/trinary patterns for effective pedestrian detection, in: Proc Asian Conference on Computer Vision, 2010, pp. 281–292.
- [63] J. Xu, Q. Wu, J. Zhang, Z. Tang, Fast and accurate human detection using a cascade of boosted MS-LBP features, IEEE Signal Processing Letters 19 (10) (2012) 676–679.
- [64] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, IEEE Transactions on Image Processing 19 (6) (2010) 1635–1650.
- [65] D. T. Nguyen, W. Li, P. Ogunbona, Local intensity distribution descriptor for object detection, IET Electronics Letters 47 (5) (2011) 321–322.

- [66] P. Ott, M. Everingham, Implicit color segmentation features for pedestrian and object detection, in: Proc International Conference on Computer Vision, 2009, pp. 723–730.
- [67] S. Walk, N. Majer, K. Schindler, B. Schiele, New features and insights for pedestrian detection, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2010, pp. 1030–1037.
- [68] P. Viola, M. J. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, *International Journal of Computer Vision* 63 (2) (2005) 153–161.
- [69] N. Dalal, B. Triggs, C. Schmid, Human detection using oriented histograms of flow and appearance, in: Proc European Conference on Computer Vision, 2006, pp. 428–441.
- [70] P. Viola, M. J. Jones, Rapid object detection using a boosted cascade of simple features, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 1, 2001, pp. 511–518.
- [71] D. T. Nguyen, P. Ogunbona, W. Li, Human detection with contour-based local motion binary patterns, in: Proc IEEE International Conference on Image Processing, 2011, pp. 3609–3612.
- [72] D. Park, C. L. Zitnick, D. Ramanan, P. Dollár, Exploring weak stabilization for motion feature extraction, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2013, pp. 2882–2889.
- [73] M. Enzweiler, A. Eigenstetter, B. Schiele, D. M. Gavrila, Multi-cue pedestrian classification with partial occlusion handling, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2010, pp. 990–997.
- [74] F. Zhou, F. D. L. Torre, Spatial-temporal matching for human detection in video, in: Proc European Conference on Computer Vision, 2014, pp. 62–77.
- [75] H. Wang, A. Kläser, C. Schmid, C. L. Liu, Dense trajectories and motion boundary descriptors for action recognition, *International Journal of Computer Vision* 103 (1) (2013) 60–79.
- [76] G. Farneback, Two-frame motion estimation based on polynomial expansion, in: Proc Scandinavian Conference on Image Analysis, 2003, pp. 363–370.
- [77] K. Levi, Y. Weiss, Learning object detection from a small number of examples: The importance of good features, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2004, pp. 53–60.
- [78] Y. T. Chen, C. S. Chen, A cascade of feed-forward classifiers for fast pedestrian detection, in: Proc Asian Conference on Computer Vision, 2007, pp. 905–914.
- [79] B. Wu, R. Nevatia, Optimizing discrimination-efficiency tradeoff in integrating heterogeneous local features for object detection, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
- [80] A. Opelt, A. Pinz, A. Zisserman, Learning an alphabet of shape and appearance for multi-class object detection, *International Journal of Computer Vision* 80 (1) (2008) 16–44.

- [81] X. Wang, T. X. Han, S. Yan, An HOG-LBP human detector with partial occlusion handling, in: Proc International Conference on Computer Vision, 2009, pp. 32–39.
- [82] J. Zhang, K. Huang, Y. Yu, T. Tan, Boosted local structured HOG-LBP for object localization, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2011, pp. 1393–1400.
- [83] C. Wojek, S. Walk, B. Schiele, Multi-cue onboard pedestrian detection, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 794–801.
- [84] P. Dollár, Z. Tu, P. Perona, S. Belongie, Integral channel features, in: Proc British Machine Vision, 2009.
- [85] R. Benenson, M. Mathias, T. Tuytelaars, L. V. Gool, Seeking the strongest rigid detector, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2013, pp. 3666–3673.
- [86] S. Walk, K. Schindler, B. Schiele, Disparity statistics for pedestrian detection: Combining appearance, motion and stereo, in: Proc European Conference on Computer Vision, Vol. 6, 2010, pp. 182–195.
- [87] A. D. Costea, S. Nedeveschi, Word channel based multiscale pedestrian detection without image resizing and using only one classifier, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2014, pp. 2393–2400.
- [88] E. Seemann, B. Leibe, B. Schiele, Multi-aspect detection of articulated objects, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 2, 2006, pp. 1582–1588.
- [89] E. Seemann, B. Schiele, Cross-articulation learning for robust detection of pedestrians, in: Proc DAGM, 2006, pp. 242–252.
- [90] K. Mikolajczyk, B. Leibe, B. Schiele, Multiple object class detection with a generative model, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 1, 2006, pp. 26–36.
- [91] Q. Zhu, S. Avidan, M. C. Yeh, K. T. Cheng, Fast human detection using a cascade of histograms of oriented gradients, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 2, 2006, pp. 1491–1498.
- [92] V. D. Shet, J. Neumann, V. Ramesh, L. S. Davis, Bilattice-based logical reasoning for human detection, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.
- [93] D. Tran, D. Forsyth, Configuration estimates improve pedestrian finding, in: Proc Conference on Neural Information Processing Systems, 2007, pp. 1–8.
- [94] S. Maji, A. C. Berg, J. Malik, Classification using intersection kernel support vector machines is efficient, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

- [95] P. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
- [96] C. H. Lampert, M. B. Blaschko, T. Hofmann, Beyond sliding windows: Object localization by efficient subwindow search, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
- [97] C. Wojek, B. Schiele, A performance evaluation of single and multi-featuer people detection, in: Proc DAGM, 2008, pp. 82–91.
- [98] Z. Lin, G. Hua, L. S. Davis, Multiple instance feature for robust part-based object detection, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 405–412.
- [99] L. Bourdev, J. Malik, Poselets: Body part detectors trained using 3D human pose annotations, in: Proc International Conference on Computer Vision, 2009, pp. 1365–1372.
- [100] T. Yeh, J. J. Lee, T. Darrell, Fast concurrent object localization and recognition, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2009, pp. 280–287.
- [101] H. Harzallah, F. Jurie, C. Schmid, Combining efficient object localization and image classification, in: Proc International Conference on Computer Vision, 2009, pp. 237–244.
- [102] W. R. Schwartz, A. Kembhavi, D. Harwood, L. S. Davis, Human detection using partial least squares analysis, in: Proc International Conference on Computer Vision, 2009, pp. 24–31.
- [103] O. Barinova, V. Lempitsky, P. Kohli, On detection of multiple object instances using Hough transforms, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2010, pp. 2233–2240.
- [104] M. Pedersoli, J. González, A. D. Bagdanov, J. J. Villanueva, Recursive coarse-to-fine localization for fast object detection, in: Proc European Conference on Computer Vision, Vol. 6, 2010, pp. 280–293.
- [105] Z. Lin, L. S. Davis, Shape-based human detection and segmentation via hierarchical part-template matching, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (4) (2010) 604–618.
- [106] P. Dollár, S. Belongie, P. Perona, The fasted pedestrian detector in the west, in: Proc British Machine Vision, 2010, pp. 1–11.
- [107] M. Enzweiler, D. M. Gavrila, Integrated pedestrian classification and orientation estimation, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2010, pp. 982–989.
- [108] R. B. Girshick, P. F. Felzenszwalb, D. A. McAllester, Object detection with grammar models, in: Proc Conference on Advances in Neural Information Processing Systems, 2011, pp. 442–450.
- [109] M. Wang, X. Wang, Automatic adaptation of a generic pedestrian detector to a specific traffic scene, in: Proc IEEE International Conference on Computer Vison and Pattern Recognition, 2011, pp. 1–12.

- [110] X. Cao, Z. Wang, P. Yan, X. Li, Rapid pedestrian detection in unseen scenes, *Neurocomputing* 74 (2011) 3343–3350.
- [111] H. Cho, P. E. Rybski, A. B. Hillel, W. Zhang, Real-time pedestrian detection with deformable part models, in: *Proc IEEE International Intelligent Vehicles Symposium*, 2012, pp. 1035–1042.
- [112] R. Xu, J. Jiao, B. Zhang, Q. Ye, Pedestrian detection in images via cascaded L1-norm minimization learning method, *Pattern Recognition* 45 (7) (2012) 2573–2583.
- [113] Y. Ding, J. Xiao, Contextual boost for pedestrian detection, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2895–2902.
- [114] W. Ouyang, X. Wang, A discriminative deep model for pedestrian detection with occlusion handling, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3258–3265.
- [115] R. Benenson, M. Mathias, R. Timofte, L. V. Gool, Pedestrian detection at 100 frames per second, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2903–2910.
- [116] Y. Xu, D. Xu, S. Lin, T. X. Han, X. Cao, X. Li, Detection of sudden pedestrian crossings for driving assistance systems, *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics* 42 (3) (2012) 729–739.
- [117] S. Tang, M. Andrikula, B. Schiele, Detection and tracking of occluded people, in: *Proc British Machine Vision Conference*, 2012, pp. 1–11.
- [118] Q. Ye, Z. Han, J. Jiao, J. Liu, Human detection in images via piecewise linear support vector machines, *IEEE Transactions on Image Processing* 22 (2) (2013) 778–789.
- [119] W. Ouyang, X. Wang, Single-pedestrian detection aided by multi-pedestrian detection, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3198–3205.
- [120] W. Ouyang, X. Zeng, X. Wang, Modeling mutual visibility relationship in pedestrian detection, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3222–3229.
- [121] X. Cao, Z. Wang, P. Yan, X. Li, Transfer learning for pedestrian detection, *Neurocomputing* 100 (2013) 51–57.
- [122] D. T. Nguyen, P. Ogunbona, W. Li, A novel shape-based non-redundant local binary pattern descriptor for object detection, *Pattern Recognition* 46 (5) (2013) 1485–1500.
- [123] D. T. Nguyen, W. Li, P. Ogunbona, Inter-occlusion reasoning for human detection based on variational mean field, *Neurocomputing* 110 (2013) 51–61.
- [124] X. Zeng, W. Ouyang, X. Wang, Multi-stage contextual deep learning for pedestrian detection, in: *Proc International Conference on Computer Vision*, 2013, pp. 121–128.
- [125] J. Marín, D. Vázquez, A. M. López, J. Amores, B. Leibe, Random forests of local experts for pedestrian detection, in: *Proc International Conference on Computer Vision*, 2013, pp. 2592–2599.

- [126] S. Tang, M. Andrikula, A. Milan, K. Schindler, S. Roth, B. Schiele, Learning people detectors for tracking in crowded scenes, in: Proc International Conference on Computer Vision, 2013, pp. 1049–1056.
- [127] Y. Yang, D. Ramanan, Articulated human detection with flexible mixtures-of-parts, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (12) (2013) 2878–2890.
- [128] J. Yan, X. Zhang, Z. Lei, S. Liao, S. Z. Li, Robust multi-resolution pedestrian detection in traffic scenes, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2013, pp. 3033–3040.
- [129] G. Chen, Y. Ding, J. Xiao, T. X. Han, Detection evolution with multi-order contextual co-occurrence, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2013, pp. 1798–1805.
- [130] C. Yao, X. Bai, W. Liu, L. J. Latecki, Human detection using learned part alphabet and pose dictionary, in: Proc European Conference on Computer Vision, 2014.
- [131] J. Marín, D. Vázquez, A. M. López, J. Amores, L. I. Kuncheva, Occlusion handling via random subspace classifiers for human detection, IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics 44 (3) (2014) 342–354.
- [132] X. Wang, M. Wang, W. Li, Scene-specific pedestrian detection for static video surveillance, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (2) (2014) 361–374.
- [133] P. Dollár, R. Appel, S. Belongie, P. Perona, Fast feature pyramids for object detection, IEEE Transactions on Pattern Analysis and Machine Intelligence (2014) 1–14.
- [134] P. Luo, Y. Tian, X. Wang, X. Tang, Switchable deep network for pedestrian detection, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2014, pp. 899–906.
- [135] G. Gkioxari, B. Hariharan, R. Girshick, J. Malik, Using k-poselets for detecting people and localizing their keypoints, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2014, pp. 3582–3589.
- [136] J. Xu, S. Ramos, D. Vázquez, A. M. López, Incremental domain adaptation of deformable part-based models, in: Proc British Machine Vision Conference, 2014, pp. 1–12.
- [137] X. Zeng, W. Ouyang, M. Wang, X. Wang, Deep learning of scene-specific classifier for pedestrian detection, in: Proc European Conference on Computer Vision, 2014, pp. 472–487.
- [138] B. Dayer, T. Brox, Training deformable object models for human detection based on alignment and clustering, in: Proc European Conference on Computer Vision, 2014, pp. 406–420.
- [139] W. Förstner, E. Gülch, A fast operator for detection and precise location of distinct points, in: Intercommission Conference on Fast Processing of Photogrammetric Data, 1987, pp. 281–305.
- [140] C. Harris, M. Stephens, A combined corner and edge detector, in: Alvey Conference, 1988, pp. 189–192.

- [141] D. G. Lowe, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision* 60 (2) (2004) 91–110.
- [142] P. F. Felzenszwalb, D. P. Huttenlocher, Pictorial structures for object recognition, *International Journal of Computer Vision* 61 (1) (2005) 55–79.
- [143] X. Jiang, Asymmetric principal component and discriminant analyses for pattern classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (5) (2009) 931–937.
- [144] A. Criminisi, J. Shotton, E. Konukoglu, Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning, Tech. rep., Microsoft Research (2011).
- [145] S. Munder, D. M. Gavrilu, An experimental study on pedestrian classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (11) (2006) 1863–1868.
- [146] O. Maron, T. L. Pérez, A framework for multiple instance learning, in: *Proc Conference on Advances in Neural Information Processing Systems*, 1998, pp. 570–576.
- [147] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (5) (2002) 603–619.
- [148] T. K. Ho, The random subspace method for constructing decision forests, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (8) (1998) 832–844.
- [149] G. E. Hinton, S. Osindero, Y. Teh, A fast learning algorithm for deep belief nets, *Neural Computation* 18 (2006) 1527–1554.
- [150] D. Cheda, D. Ponsa, A. M. López, Pedestrian candidates generation using monocular cues, in: *Proc IEEE International Intelligent Vehicles Symposium*, 2012, pp. 7–12.
- [151] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, K. Siddiqi, TurboPixels: Fast superpixels using geometric flows, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (12) (2009) 2290–2297.
- [152] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (VOC) challenge, *International Journal of Computer Vision* 88 (2) (2010) 303–338.
- [153] B. C. Russell, A. Torralba, K. P. Murphy, W. T. Freeman, LabelMe: a database and web-based tool for image annotation, *International Journal of Computer Vision* 77 (1-3) (2008) 157–173.
- [154] S. Agarwal, D. Roth, Learning a sparse representation for object detection, in: *Proc European Conference on Computer Vision*, 2002, pp. 113–130.
- [155] P. Sermanet, K. Kavukcuoglu, S. Chintala, Y. LeCun, Pedestrian detection with unsupervised multi-stage feature learning, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3626–3633.
- [156] X. Ren, D. Ramanan, Histograms of sparse codes for object detection, in: *Proc IEEE International Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3246–3253.
- [157] C. Szegedy, A. Toshev, D. Erhan, Deep neural networks for object detection, in: *Proc Conference on Advances in Neural Information Processing Systems*, 2013, pp. 2553–2561.

- [158] D. Erhan, C. Szegedy, A. Toshev, D. Anguelov, Scalable object detection using deep neural networks, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2014, pp. 2155–2162.
- [159] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.
- [160] M. Y. Liu, O. Tuzel, A. Veeraraghavan, R. Chellappa, Fast directional chamfer matching, in: Proc IEEE International Conference on Computer Vision and Pattern Recognition, 2010, pp. 1696–1703.