Human Detection Using Learned Part Alphabet and Pose Dictionary

Cong Yao[1], Xiang Bai[1,⋆], Wenyu Liu[1], and Longin Jan Latecki[2]

[1] Department of Electronics and Information Engineering,
Huazhong University of Science and Technology
[2] Department of Computer and Information Sciences, Temple University
yaocong2010@gmail.com, {xbai,liuwy}@hust.edu.cn, latecki@temple.edu

Abstract. As structured data, human body and text are similar in many aspects. In this paper, we make use of the analogy between human body and text to build a compositional model for human detection in natural scenes. Basic concepts and mature techniques in text recognition are introduced into this model. A discriminative alphabet, each grapheme of which is a mid-level element representing a body part, is automatically learned from bounding box labels. Based on this alphabet, the flexible structure of human body is expressed by means of symbolic sequences, which correspond to various human poses and allow for robust, efficient matching. A pose dictionary is constructed from training examples, which is used to verify hypotheses at runtime. Experiments on standard benchmarks demonstrate that the proposed algorithm achieves state-of-the-art or competitive performance.

Keywords: Human detection, mid-level elements, part alphabet, pose dictionary, matching.

1 Introduction

Human detection in natural images has been an active research topic for decades and has attracted continuous attention from the computer vision community [21,32,1,5,34,41,10]. Though considerable progress has been made in recent years [8,13,20,23], detecting people in uncontrolled environments remains a challenging task. Human pose articulation, scale change, partial occlusion, low resolution, varied illumination, and complex background all constitute major challenges to human detection.

To tackle these challenges, a rich body of research has been devoted, among which part-based methods [1,11,22,20,4] have become increasingly popular in this field, due to their advantage in handling pose variation and partial occlusion. In this paper, we investigate the problem of human detection from a different perspective and propose a novel part-based human detection algorithm.

The algorithm is motivated by the key observation that human body and text are similar in many aspects. Notably: (1) They both consist of a set of basic

⋆ Corresponding author.

D. Fleet et al. (Eds.): ECCV 2014, Part V, LNCS 8693, pp. 251–266, 2014.

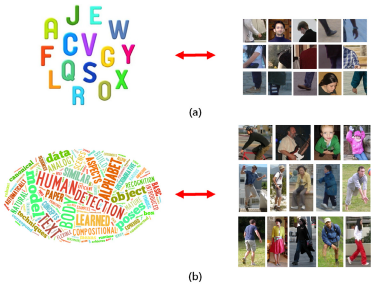

Fig. 1. Analogy between human body and text. (a) Letters *versus* parts. (b) Words *versus* poses.

primitives. For text, these basic primitives are letters in the alphabet, while for human body they are parts, such as head, shoulder, waist, and foot (Fig. 1 (a)). (2) They may exhibit significant variability. The type, number of primitives and their spatial relation are all very crucial as they jointly determine the expression of a particular object. The variation in the type and number of primitives and their spatial relation may lead to highly diverse expressions (different words in text and various poses in human body, as shown in Fig. 1 (b)). (3) They are both structured objects. The spatial relation of the primitives are not random, but instead with high degree of regularity. For example, in English text the letter 't' is very likely to be followed by 'h'. For human body, the position of head is tightly coupled with that of shoulders.

Since there are well established and widely used concepts and techniques in text recognition, we can make use of the analogy between human body and text and transfer some basic concepts and mature techniques in text recognition to the domain of human detection.

While an alphabet already exists for text, there is no visual alphabet for human in natural images. Therefore, an alphabet for human parts should be learned automatically from training data. In this paper, the discriminative clustering algorithm proposed by Singh et al. [36] is employed to learn the part alphabet.

Having learned the part alphabet, we are able to build a representation for human body. Similar to words in text, which are strings consisting of different types and numbers of letters, human poses can be represented in the form of sequences of parts. This representation converts the 3D structure of human body into 1D sequence. Though information loss is inevitable in this conversion, the ingredients of human body and their relationship are mostly preserved. Similar human poses have similar sequences while different human poses correspond to dissimilar sequences.

The benefit of representing human poses by sequences is that comparison and matching of human poses can be transformed into string matching [30], which has been proven to be both robust and efficient in text recognition. For human detection, hypothesis verification can be accomplished by matching the hypotheses with a set of reference poses. In this paper, the reference poses are given in the

training data and are converted into a collection of symbolic sequences, which we call *pose dictionary*.

In summary, the contributions of this paper are three-fold: (1) We exploit automatically learned mid-level primitives to represent human parts; (2) We propose to express human poses using symbolic sequences; and (3) We employ string matching in text recognition to perform hypothesis verification for human detection.

To assess the performance of the proposed algorithm, we have conducted experiments on standard benchmarks. It is demonstrated that the proposed algorithm achieves state-of-the-art or competitive performance, compared to other competing methods.

The rest of the paper is organized as follows. Sec. 2 reviews related works in this field. We describe the details of the proposed method, including the procedure of learning part alphabet and pose dictionary as well as the pipeline of human detection, in Sec. 3. Sec. 4 summarizes the proposed algorithm and discusses its connections to existing methods. Sec. 5 presents experimental results. Conclusion remarks and future work are given in Sec. 6.

2 Related Work

As one of the most competitive domains in computer vision, human detection has attracted quite a lot of attention from the community [8,1,43,4,10,39]. Comprehensive surveys on this topic can be found in [17,15].

The analogy between text and visual data has inspired a number of researchers in the computer vision community. Basic ideas and models for processing text have been adapted to perform vision tasks. For example, Sivic et al. [37] proposed an approach to object matching in videos by recasting the problem as text retrieval; Fei-Fei et al. [19] adopted Bag-of-Words to represent images and perform scene categorization. In this paper, we learn an alphabet to represent human body parts and build a dictionary to characterize human poses. Furthermore, string matching [30], a technique widely used in text recognition and retrieval, is employed to verify hypotheses in human detection.

The work presented in this paper is also inspired by the discriminative clustering approaches proposed by Singh et al. [36] and Lee et al. [25]. In the algorithm of Singh et al., a set of representative patch clusters is automatically discovered from a large image set. The discovered patch clusters are mid-level representation for natural images, which can be used for a wide rang of tasks such as scene classification [36] and geographically-informed image retrieval [9]. We adopt this algorithm to learn part prototypes for human body.

In [31], Opelt et al. proposed to learn a visual alphabet of shape and appearance to represent and detect objects. Our approach is different from this algorithm in: (1) the type of local descriptors (HOG descriptors on patches *vs.* boundary fragments or SIFT descriptors on interest points); (2) the usage of alphabet graphemes (strong detectors *vs.* weak detectors); and (3) the manner of hypothesis verification (string matching *vs.* Adaboost classifier).

Fig. 2. Part alphabet generation. (a) Learned alphabet on the TUD-Pedestrians dataset [1]. Each row illustrates a cluster of part instances that constitute a grapheme in the alphabet. The images in the first column (orange rectangle) are the average of all the instances of that grapheme. The rest are top-ranked part instances. (b) Discovered part instances in original images. The learned part prototypes are tightly clustered in both appearance and configuration space. (c) Learned alphabet on the INRIA Person dataset [8].

Similar to our work, Andriluka et al. [1] and Bourdev et al. [5,4] learn part-based models to detect people in natural scenes. However, the part prototypes in their algorithms are obtained using detailed annotations of body parts, while in our model the part prototypes are inferred automatically without part annotations.

Wang et al. [42] presented a framework for discovering salient object parts, which was shown to be robust to object articulation. However, the framework used fixed number of parts and implicitly assumed common structure among different object instances, which can be violated in case of viewpoint change or occlusion. In contrast, in our model different object instances are represented in variable number of parts, and pose variation, viewpoint change and occlusion are treated as variants of symbolic sequences.

The proposed method shares the idea of predicting object centroid via Hough voting with [26] and [22], but the representation and detection pipeline are different from those of [26] and [22].

Most related to our work, the algorithm of Endres et al. [16] also learned mid-level elements to represent object parts. However, our work is different from [16] in that it learns the part detectors jointly while Endres et al. trained part detectors independently. Moreover, our part detectors are more efficient at runtime as the parts are described by HOGs with the same resolution and thus allow highly parallelized part detection.

3 Methodology

3.1 Part Alphabet Generation

Given a set of training images of humans $S = \{(I_i, B_i)\}_{i=1}^{n}$, where I_i is an image and B_i is a set of bounding boxes specifying the location and extent of the humans in the image I_i ($B_i = \emptyset$, if I_i is person-free), the goal of part alphabet generation is to learn a set of part prototypes Ω from S. The part prototypes

should have two properties: (1) Representativeness. The prototypes should be able to capture the essential sub-structures of human body and be common across different human poses, at least for similar poses. (2) Discrimination. The prototypes should be distinctive from background and against each other, otherwise, there will be tremendous confusion and ambiguity when applying them to novel images, which will make the detection of human in natural images fail.

Since only bounding box annotations are available in S, these part prototypes should be automatically discovered. The discriminative clustering algorithm proposed by Singh et al. [36] meets the requirements well, as it can discover visual primitives that are both representative and discriminative from large image collections in an unsupervised manner. Inspired by [36] and [44], we adopt this algorithm to learn the part alphabet Ω from S.

Given a "discovery" image set $\mathcal{D}$ and a "natural world" image set $\mathcal{N}$, the algorithm of Singh et al. [36] is aimed to discover a set of representative patch clusters that are discriminative against other clusters in $\mathcal{D}$, as well as the rest visual world modelled by $\mathcal{N}$. The algorithm is an iterative procedure which alternates between two phases: clustering and training. Initially, examples (patches) are grouped into clusters in an unsupervised fashion and then a discriminative classifier is trained for each cluster using the patches in the cluster as positive examples and the rest as negative examples. In next iteration, these classifiers are used to find patches similar to those in the corresponding cluster in novel images, which is followed by a new round of training. The algorithm iterates until convergence.

The output of the algorithm is a set of top-ranked patch clusters K and a set of classifiers C. Each cluster K_j corresponds to a classifier C_j that can detect patches similar to those in K_j in novel images. These classifiers will serve as part detectors at runtime.

The algorithm of Singh et al. [36] was originally designed for discovering discriminative patches from generic natural images. To utilize it to discover part prototypes from training examples, we made the following customizations:

- The regions in the bounding boxes B constitute the discovery set $\mathcal{D}$ as we aim to discover discriminative parts for human and the rest regions of the training image set I are taken as the natural world set $\mathcal{N}$.
- At the initial clustering stage, each patch p_k from the discover set is represented by a location-augmented descriptor, which is the concatenation of the appearance descriptor and the normalized coordinates (x_{p_k}, y_{p_k}), following [29]. This makes the patches in each cluster more compact in configuration space.
- The scale of the patches (following [36], we also use square patches, i.e. the width w and height h are equal and $w = h = s$) sampled from the discovery set is adaptive to the scale of the bounding box bb. The scale of a specific patch is $s = r \cdot max(w(bb), h(bb))$, where $r \in (0, 1]$ is scale ratio which controls the relative scale of the patches.
- To make the learned parts distinctive from background cluster, we also randomly draw examples from the natural world set $\mathcal{N}$ at different scales.

- To make the trained classifiers more robust to scale change, the training set is enriched by rescaling the original images at multiple scales.
- The SVM classifier used in [36] was replaced by Random Forest [6] because Random Forest can achieve similarly high accuracy as SVM and directly gives probabilities, which are more intuitive and interpretable.
- The size of the patch descriptors (HOG [8]) is 3×3 (rather than 8×8) cells as they are sufficient for describing local body parts.

The learned part alphabet can be expressed as $\Omega = \{(K_j, C_j)\}_{j=1}^{\Gamma}$, where K and C are the discovered part prototypes and corresponding classifiers respectively, and Γ is the size of the alphabet. For each cluster K_j, the following information is stored: The set of all its members (patches) M_j, their offset vectors to object centroid V_j, and the average width $\bar{w}_j$ and height $\bar{h}_j$ of the parent rectangles, from which the members M_j originate. V_j, $\bar{w}_j$ and $\bar{h}_j$[1] will be used to estimate the location and extent of objects in the detection phase (see Sec. 3.3).

Fig. 2 depicts the alphabets (classifiers not shown) learned on the TUD-Pedestrians [1] and INRIA Person [8] dataset. The learned part prototypes are tightly clustered in both appearance and configuration space (Fig. 2 (b)), which are very much in common with poselets [5,4]. However, different from poselets, which are obtained using manually labeled part regions and keypoints, our part prototypes are automatically learned using human bounding boxes.

As shown in Fig. 2 (b), the learned part prototypes do not necessarily correspond to single semantic body part. For example, the part prototype in the bottom row fires on both left and right foot. However, this is reasonable as the patches are very similar in both appearance and configuration space. More importantly, the learned parts are sufficient for the task of human detection and work well in practice (see the experiments in Sec. 5).

3.2 Pose Dictionary Construction

Having learned an alphabet for representing human body parts, we are now able to construct a dictionary to describe human poses. The procedure of pose dictionary construction is illustrated in Fig. 3.

For each positive example in the training set, part detection is performed within the bounding box using the trained part detectors. In accordance with the alphabet generation stage, the scale of the detection windows is $s = r \cdot max(w(bb), h(bb))$. Non-maximum suppression is applied to the detection activations to eliminate redundancy. The scores of different part detectors are directly comparable as they are trained in a one-versus-all manner.

The detected parts are then sorted by the azimuth relative to the body center (yellow cross in Fig. 3). The azimuth angle of each part is measured clockwise from a north base line (red arrow in Fig. 3). A one-dimensional sequence is formed by successively recording the indices of the parts after sorting (orange

[1] We assume that V_j, $\bar{w}_j$ and $\bar{h}_j$ have been normalized with respect to the members M_j.

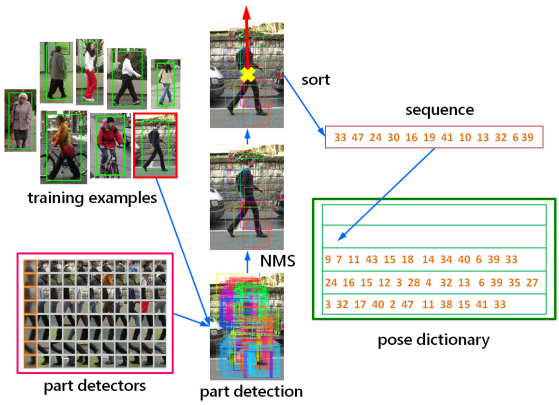

Fig. 3. Pose dictionary construction. Parts are detected by applying the trained part detectors to the positive training examples (different parts are marked in different colors). The detected parts are then sorted by the azimuth relative to the body center to convert a pose to a one-dimensional sequence. Due to variation or partial occlusion, different poses may correspond to sequences of variable lengths.

numbers in Fig. 3). The sequence is appended to the pose dictionary, which will be used in the detection phase to certificate human hypotheses. The production of the pose dictionary construction procedure is $\Phi = \{\phi_l\}_{l=1}^{\Pi}$, where each ϕ_l is a one-dimensional sequence that represents a pose in the training set and Π stands for the size of the dictionary.

To make the representation more robust, random jittering is applied to the starting point of the original sequence to generate multiple sequences for each training example.

The variability caused by pose variation and viewpoint change is implicitly encoded by the symbolic sequences. More importantly, in our model different poses are expressed in different number of parts (in contrast to fixed number of parts in [20]), which yields a more flexible representation for modelling articulated objects. More sophisticated approaches that are able to capture the 2D (or even 3D) nature of human body can be incorporated, however, the current strategy is already quite effective and efficient.

3.3 Detection Pipeline

Generally, the proposed detection pipeline works in a hypothesis generation and verification paradigm [40]. We follow up traditional object detection methods to search human instances in images in a multi-scale sliding-window manner and fuse activations of different locations and scales to form the final detections. In the detection phase, the images are fixed and windows of multiple scales are densely sampled and fed to the part detectors. In the following paragraphs, we present the processes of hypothesis generation and verification in an image at a single scale for simplicity.

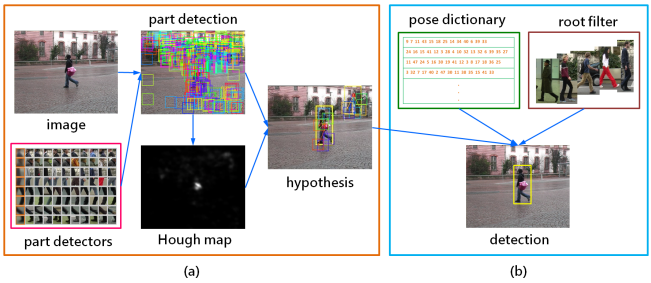

Fig. 4. Human detection at a single scale. (a) Hypothesis generation. (b) Hypothesis verification. See text for details.

Hypothesis Generation. Patches are densely sampled and described by HOG descriptors with 3×3 cells. Parts are detected using the learned detectors C. A Hough map is then generated by casting and accumulating the votes from the detected parts, similar to [26]. The vote of each part is the highest score the part receives from the detectors C. Centers of hypotheses are found by seeking maxima in the Hough map using Mean Shift [7].

For each hypothesis h, back-projection is performed to seek the parts (denoted as a set $Q(h)$) that have contributed to h. Non-maximum suppression is applied to $Q(h)$, to remove redundant parts, resulting in $Q^{'}(h)$. A sequence $\psi(h)$ is formed using $Q^{'}(h)$ in the same way as in the pose dictionary construction procedure.

The width and height of hypothesis h is estimated using the corresponding clusters the parts in $Q(h)$ belong to:

$$w(h) = \frac{\sum_l \rho(Q_l) \cdot w(Q_l) \cdot \bar{w}_{Q_l}}{\sum_l \rho(Q_l)}, \tag{1}$$

$$h(h) = \frac{\sum_l \rho(Q_l) \cdot h(Q_l) \cdot \bar{h}_{Q_l}}{\sum_l \rho(Q_l)}, \tag{2}$$

where $\rho(Q_l)$ is the detection score of Q_l, $w(Q_l)$ and $h(Q_l)$ stand for the width and height of Q_l, and $\bar{w}_{Q_l}$ and $\bar{h}_{Q_l}$ denote the average width and height of the cluster corresponding to Q_l, respectively.

The total vote of hypothesis h, $\alpha(h)$, is also calculated as follows: $\alpha(h) = \sum_l \rho(Q_l)$. We use total vote instead of mean vote as low level evidence, as hypotheses formed by spurious parts may have high mean vote, therefore using mean vote may lead to confusion.

Hypothesis Verification via Dictionary Search. Dictionary search [24,30] is a popular technique for error correction in text recognition. Basically, dictionary search tries to find the closest match(es) in the dictionary for a given string. The similarity between the input string and the matched entry (or entries) can be used to verify whether the input string is erroneous. We adopt this method to verify hypotheses, since they have been expressed in sequences.

The edit distance [27] is the most widely used technique in dictionary search, which can efficiently compute the distance (dissimilarity) of two strings (sequences). In the edit distance, three basic operations are allowable: insertion, deletion, and substitution. Interesting correspondence can be observed between these operations and part-based image matching: insertion corresponds to part missing, deletion corresponds to spurious part and substitution to incorrect part type. This correspondence makes the edit distance [27] particularly suitable for matching and verifying hypotheses for human detection, as it can tolerate errors in the bottom-up hypothesis generation stage.

Formally, given two sequences ψ and ϕ with edit distance $d(\psi, \phi)$, the normalized distance [28] is defined as:

$$\hat{d}(\psi, \phi) = \frac{2d(\psi, \phi)}{L(\psi) + L(\phi) + d(\psi, \phi)}, \tag{3}$$

where $L(\psi)$ and $L(\phi)$ denote the lengths of ψ and ϕ.

For a hypothesis h expressed in sequence $\psi(h)$, T closest entries $\{\phi_i\}_{i=1}^{T}$ in the dictionary Φ are searched. The score of hypothesis h via dictionary search, $\beta(h)$, is defined as the average similarity:

$$\beta(h) = \frac{1}{T} \sum_{i=1}^{T} (1 - \hat{d}(\psi(h), \phi_i)). \tag{4}$$

$\beta(h)$ measures the possibility of hypothesis h being a valid human pose.

Hypothesis Verification via Root Filter. Merely considering local parts may lose information from global structure of object, thus we also train a root filter [20] as compensation. The examples for training the root filter Υ are harvested by applying hypothesis generation to the training images and comparing the bounding boxes of the generated hypotheses and ground truth rectangles. A hypothesis is taken as positive example if it overlaps significantly with a ground truth rectangle (overlap ratio ≥ 0.5). The sub images within the bounding boxes are normalized and represented by HOG descriptors.

To deal with pose variation and viewpoint change, multiple components are introduced into the root filter, following [20]. The components are formed by clustering the training examples according to their aspect ratio. The optimal value of component number m depends on the variability of objects.

A Random Forest classifier [6] is trained for each component using the harvested examples. In the verification phase, for each hypothesis h the component with proximal aspect ratio is used to predict the probability of h being an object and this probability serves as the output of the root filter: $\gamma(h) = \Upsilon(h)$.

Metric Fusion. As described above, for each hypothesis h there are three metrics that measure the possibility of h representing a true object: $\alpha(h)$ stands for the local evidence from part detection; $\beta(h)$ characterizes the interactions among the parts of h; and $\gamma(h)$ induces global information. These metrics should be fused to give a unique score for h. However, since not all the three metrics are

at the same scale ($\alpha(h)$ is the sum of multiple votes), simple linear combination will lead to poor result.

In this paper, we use the harmonic mean [33] for metric fusion, as it can handle metrics at different scales. The final score of hypothesis h is defined as:

$$\theta(h) = \frac{3}{\frac{1}{\alpha(h)} + \frac{1}{\beta(h)} + \frac{1}{\gamma(h)}} \tag{5}$$

4 Reflection

The proposed algorithm has many interesting connections to existing methods, which we briefly discuss below.

Deformable Part Model. In Deformable Part Model [20], a star-structured model with a root template and several part templates is designed to represent objects. In the detection procedure, the location of root template is first determined and the optimal placement of the part templates with respect to the root template are then searched. In our method, parts are detected and grouped together to form a global object hypothesis and deformation and articulation are verified by a set of reference poses. In this sense, our algorithm can be seen as a bottom-up deformable part model.

Grammar Model. The pioneer work of Zhu et al. [45] established a general grammar framework for images. Following this work, Girshick et al. proposed a Grammar Model [23], which defines formal grammar for people and utilizes a compositional hierarchy that provides choices between different part subtypes and allows for optional parts, to adapt to different poses and levels of visibility. In our model, variabilities caused by pose variation and occlusion, which the grammar in [23] aims to model, are implicitly reflected in the variance of the symbolic sequences.

Poselets. Poselets [5,4] are part primitives that are by construction tightly clustered in both appearance and configuration space, for representing and detecting people. At runtime, instances of poselets are found and combined to predict location and extent of humans. The proposed algorithm works in a similar way. But the key difference is that the primitives in the proposed algorithm are learned automatically without part annotations.

5 Experiments

We have evaluated the proposed algorithm on several standard benchmarks for human detection and compared it to other competing methods, including the leading algorithms in this field. We followed the evaluation criteria for each of the datasets used in previous works. All the experiments were conducted on a regular PC (2.8GHz 8-core CPU, 16G RAM and Windows 64-bit OS).

For all the Random Forest classifiers, 100 trees were used. $T = 5$ entries in the dictionary were sought in hypothesis verification. Detection windows were sampled at 10 scales to handle size variation of humans.

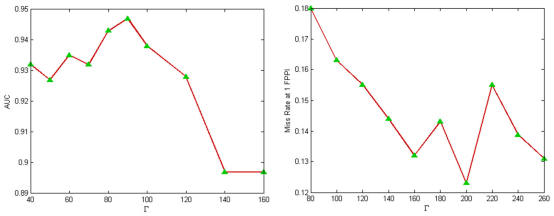

Fig. 5. Impact of alphabet size Γ on (*left*) TUD-Pedestrians and (*right*) INRIA Person

5.1 Datasets

TUD-Pedestrians. The TUD-Pedestrians dataset was proposed by Andriluka et al. in [1] and has become a widely used benchmark for assessing human detection algorithms. This database includes 250 test images of street scenes containing 311 side-view pedestrians with variability in pose, appearance and scale. As the backgrounds in the training set are with limited diversity, we also used the background images from the INRIA Person dataset [8], following [22].

INRIA Person. The INRIA Person dataset [8] is also a popular benchmark for pedestrian detection. This dataset is challenging because of pose articulation, scale change, partial occlusion, varying illumination and complex background clutter. There are 614 images with humans and 1218 person-free images for training. 741 images are used for testing. We evaluated the proposed algorithm on full images and reported per-image instead of per-window performance on this database, following [13,15].

As objects in different datasets exhibit different degrees of variability, the value of component number m varies for each dataset. In this paper, we set $m = 2, 3$ for TUD-Pedestrians and INRIA Person respectively.

5.2 Experimental Results

Scale ratio r is a crucial parameter as it determines the relative scale of the part prototypes in the alphabet. We investigated the impact of r on the TUD-Pedestrians dataset. As shown in Tab. 1, $r = 0.2$ leads to the best performance. Upon inspection, we found that too small parts only capture simple primitives like bars and corners and thus omit the characteristics of human body, while too large parts generalize poorly to novel images. Similar trend was also observed on the INRIA Person dataset, so r is fixed at 0.2 for all the following experiments.

We experimented with different alphabet sizes on the TUD-Pedestrians and INRIA Person dataset. As can be seen from Fig 5, the accuracy increases with

Table 1. Impact of scale ratio r (with $\Gamma = 80$)

r	0.1	0.2	0.3	0.4	0.5	0.6
AUC	0.608	0.943	0.596	0.133	0.043	0.009

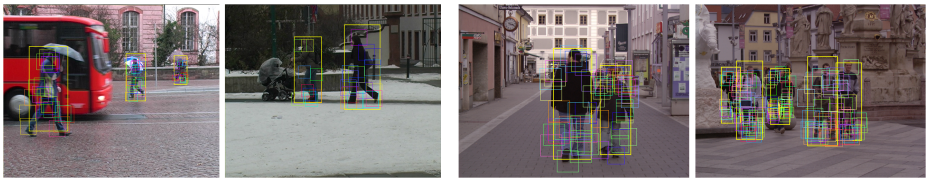

Fig. 6. Detection examples on (*left*) TUD-Pedestrians and (*right*) INRIA Person. Note that the type and number of parts vary across different human instances, as we use more flexible representation and model to represent objects, which make our algorithm different from other part-based methods, such as [1,20].

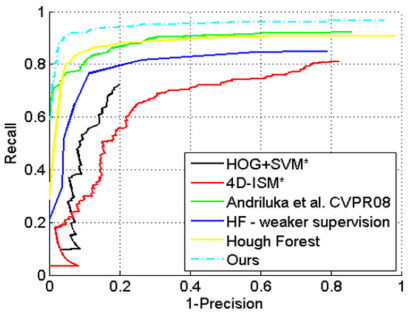

Fig. 7. Performance curves of different algorithms evaluated on the TUD-Pedestrians dataset [1]

alphabet size Γ upto a certain point and then falls. Excessive part prototypes may include redundancy and thus hurt the accuracy. The performance is not sensitive to alphabet size, as long as sufficient part prototypes are learned. Optimal result was obtained with $\Gamma = 90$ for TUD-Pedestrians and $\Gamma = 200$ for INRIA Person.

Fig. 6 depicts several detection examples of our method on the TUD-Pedestrians and INRIA Person dataset. The proposed algorithm is able to detect people of different poses and sizes under varying illumination and complex background.

Table 2. Performances of different methods evaluated on the TUD-Pedestrians dataset [1]

Algorithm	**Recall at EER**	**Detection Rate**
Ours	**0.920**	**0.965**
Hough Forest [22]	0.87	0.91
PartISM [1]	0.84	0.92
Feature Context [42]	0.73	0.84
4D-ISM [35]	0.69	0.81
HOG [8]	-	0.71

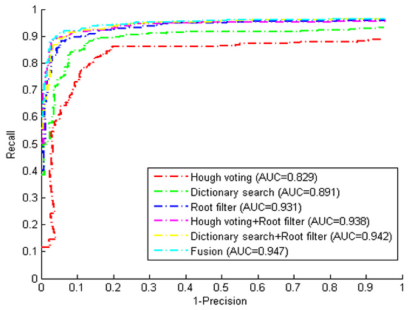

Fig. 8. Comparison of detection accuracy with different metrics and their combination

The quantitative results of different methods evaluated on the TUD-Pedestrians dataset are shown in Fig. 7 and Tab. 2. On this dataset, the proposed algorithm achieves AUC = 0.947 and recall-precision EER = 0.92, which outperforms all the competing algorithms by a large margin, including the state-of-the-art methods [1,22]. Note that in TUD-Pedestrians the test images are much more challenging than the images for training, as the variation in human pose and scale in the test images are more significant and the backgrounds are relatively more complex. This indicates that the proposed algorithm generalizes well to novel images, even though trained on simpler examples.

Without the extra negative images, our method still performs fairly well. The AUC is 0.84, on par with [1], which required detailed part annotations, and comparable to [22], which used those negative images.

Table 3. Performances of different methods evaluated on the INRIA Person dataset [1]

Algorithm	**Miss Rate at 1 FPPI**
Ours	0.12
Very Fast [3]	**0.07**
FPDW [12]	0.09
DPM-V2 [20]	0.09
Integral Channel Features [13]	0.14
HOG-LBP [43]	0.14
HOG [8]	0.23

The performances of the proposed algorithm and other competing methods on the INRIA Person dataset are depicted in Tab. 3. The proposed algorithm achieves miss rate of 0.12 at 1 false positive per image (FPPI), which is better than the traditional methods such as HOG [8], HOG-LBP [43] and Integral Channel Features [13], but still behind the best performers on this dataset [12,20,3]. The comparisons are fair, since those methods were also evaluated on full images.

We also investigated the effect of metric fusion on the TUD-Pedestrians dataset. The performances of different metrics (three types of cues from Hough voting, dictionary search and root filter) and their combinations are shown in Fig. 8. The three metrics used in isolation already lead to considerably good performance, among which root filter performs best. Hough voting and dictionary search indeed lead to further improvement. The optimal accuracy is achieved when all the cues are integrated.

On the surface, root filter provides bulk of the detection rate. However, part detection also implicitly contributes to it, since the hypotheses fed to root filter are a sparse set of bounding boxes estimated from detected parts; string matching further punishes invalid poses. Hence the considerable overall improvement is due to all 3 techniques.

On the TUD-Pedestrians dataset, the average processing time of the proposed algorithm is about 6 seconds[2], which is comparable to that of Hough Forest [22].

6 Conclusions and Future Work

We have presented a compositional model for human detection in natural scenes, which incorporates basic concepts (alphabet and dictionary) and mature techniques (edit distance and dictionary search) from text recognition. Specifically, a discriminative alphabet is learned to represent body parts. To characterize the flexible structure of human body, human poses are represented by one-dimensional sequences, which allow for robust and efficient matching. Experiments on standard benchmarks demonstrate that the proposed algorithm achieves state-of-the-art or competitive performance.

In this paper, we only demonstrated the strength of the proposed algorithm on the problem of human detection on moderate-sized datasets. Assessing the proposed algorithm on larger and more challenging datasets (such as the PASCAL VOC 2007 dataset [18] and Caltech Pedestrian Dataset [14]) is an ongoing work. The proposed model is actually quite general, thus it can be readily generalized to other object classes. We plan to build a universal model for multi-class object detection [31,20] in the future. Moreover, this work can be extended by learning part prototypes with different aspect ratios [38] and exploring the 2D/3D nature of object structure [2].

Acknowledgements. This work was primarily supported by National Natural Science Foundation of China (NSFC) (No. 61222308), and in part by NSFC (No. 61173120 and 60903096), Program for New Century Excellent Talents in University (No. NCET-12-0217) and Fundamental Research Funds for the Central Universities (No. HUST 2013TS115). This work was also supported by National Science Foundation under Grants OIA-1027897 and IIS-1302164. The authors would like to thank Xinggang Wang for the enlightening discussions and valuable suggestions.

[2] 6 threads are used to accelerate the process of part detection.

References

1. Andriluka, M., Roth, S., Schiele, B.: People-tracking-by-detection and people-detection-by-tracking. In: Proc. CVPR (2008)
2. Bai, X., Wang, X., Latecki, L.J., Liu, W.: Active skeleton for non-rigid object detection. In: Proc. ICCV (2009)
3. Benenson, R., Mathias, M., Timofte, R., Gool, L.V.: Pedestrian detection at 100 frames per second. In: Proc. CVPR (2012)
4. Bourdev, L., Maji, S., Brox, T., Malik, J.: Detecting people using mutually consistent poselet activations. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 168–181. Springer, Heidelberg (2010)
5. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: Proc. ICCV (2009)
6. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
7. Cheng, Y.: Mean shift, mode seeking, and clustering. IEEE Trans. PAMI 17(8), 790–799 (1995)
8. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR (2005)
9. Doersch, C., Singh, S., Gupta, A., Sivic, J., Efros, A.A.: What makes paris look like paris? ACM Trans. Graphics 31(3), 101 (2012)
10. Dollár, P., Appel, R., Kienzle, W.: Crosstalk cascades for frame-rate pedestrian detection. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 645–659. Springer, Heidelberg (2012)
11. Dollár, P., Babenko, B., Belongie, S., Perona, P., Tu, Z.: Multiple component learning for object detection. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 211–224. Springer, Heidelberg (2008)
12. Dollar, P., Belongie, S., Perona, P.: The fastest pedestrian detector in the west. In: Proc. BMVC (2010)
13. Dollar, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features. In: Proc. BMVC (2009)
14. Dollar, P., Wojek, C., Appel, R., Perona, P.: Pedestrian detection: A benchmark. In: Proc. CVPR (2009)
15. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: An evaluation of the state of the art. IEEE Trans. PAMI 34(4), 743–761 (2012)
16. Endres, I., Shih, K.J., Jiaa, J., Hoiem, D.: Learning collections of part models for object recognition. In: Proc. CVPR (2013)
17. Enzweiler, M., Gavrila, D.M.: Monocular pedestrian detection: Survey and experiments. IEEE Trans. PAMI 31(12), 2179–2195 (2009)
18. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes (VOC) challenge. IJCV 88(2), 303–338 (2010)
19. Fei-Fei, L., Perona, P.: A bayesian heirarcical model for learning natural scene categories. In: Proc. CVPR (2005)
20. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. PAMI 32(9), 1627–1645 (2010)
21. Forsyth, D., Fleck, M.: Body plans. In: Proc. CVPR (1997)
22. Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: Proc. CVPR (2009)
23. Girshick, R., Felzenszwalb, P., McAllester, D.: Object detection with grammar models. In: Proc. NIPS (2011)

24. Kukich, K.: Techniques for automatically correcting words in text. ACM Computing Surveys 24(4), 377–439 (1992)
25. Lee, Y.J., Efros, A.A., Hebert, M.: Style-aware mid-level representation for discovering visual connections in space and time. In: Proc. ICCV (2013)
26. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV 77(1-3), 259–289 (2008)
27. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady 10(8), 707–710 (1996)
28. Li, Y., Liu, B.: A normalized levenshtein distance metric. IEEE Trans. PAMI 29(6), 1091–1095 (2007)
29. McCann, S., Lowe, D.G.: Spatially local coding for object recognition. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) ACCV 2012, Part I. LNCS, vol. 7724, pp. 204–217. Springer, Heidelberg (2013)
30. Navarro, G.: A guided tour to approximate string matching. ACM Computing Surveys 33(1), 31–88 (2001)
31. Opelt, A., Pinz, A., Zisserman, A.: Learning an alphabet of shape and appearance for multi-class object detection. IJCV 80(1), 16–44 (2008)
32. Papageorgiou, C., Poggio, T.: A trainable system for object detection. IJCV 38(1), 15–33 (2000)
33. Van Rijsbergen, C.: Information Retrieval, 2nd edn. Butterworths, London (1979)
34. Schwartz, W.R., Kembhavi, A., Harwood, D., Davis, L.S.: Human detection using partial least squares analysis. In: Proc. ICCV (2009)
35. Seemann, E., Schiele, B.: Cross-articulation learning for robust detection of pedestrians. In: Franke, K., Müller, K.-R., Nickolay, B., Schäfer, R. (eds.) DAGM 2006. LNCS, vol. 4174, pp. 242–252. Springer, Heidelberg (2006)
36. Singh, S., Gupta, A., Efros, A.A.: Unsupervised discovery of mid-level discriminative patches. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 73–86. Springer, Heidelberg (2012)
37. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: Proc. ICCV (2003)
38. Song, X., Wu, T., Jia, Y., Zhu, S.C.: Discriminatively trained and-or tree models for object detection. In: Proc. CVPR (2013)
39. Tan, D., Li, Y., Kim, T.K.: Fast pedestrian detection by cascaded random forest with dominant orientation templates. In: Proc. BMVC (2012)
40. Tsai, S.S., Parameswarany, V., Berclazy, J., Vedanthamy, R., Grzeszczuky, R., Girod, B.: Design of a text detection system via hypothesis generation and verification. In: Proc. ACCV (2012)
41. Walk, S., Majer, N., Schindler, K., Schiele, B.: New features and insights for pedestrian detection. In: Proc. ICCV (2010)
42. Wang, X., Bai, X., Yang, X., Liu, W., Latecki, L.J.: Maximal cliques that satisfy hard constraints with application to deformable object model learning. In: Proc. NIPS (2011)
43. Wang, X., Han, T.X., Yan, S.: An HOG-LBP human detector with partial occlusion handling. In: Proc. ICCV (2009)
44. Yao, C., Bai, X., Shi, B., Liu, W.: Strokelets: A learned multi-scale representation for scene text recognition. In: Proc. CVPR (2014)
45. Zhu, S.C., Mumford, D.: A stochastic grammar of images. Foundations and Trends in Computer Graphics and Vision 2(4), 259–362 (1995)