This CVPR2013 paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Detection Evolution with Multi-Order Contextual Co-occurrence

Guang Chen* Yuanyuan Ding† Jing Xiao† Tony X. Han*
†Epson Research and Development, Inc.
San Jose, CA, USA
{yding,xiaoj}@erd.epson.com
*Dept. of ECE, Univ. of Missouri
Columbia, MO, USA
gc244@mail.missouri.edu hantx@missouri.edu

Abstract

Context has been playing an increasingly important role to improve the object detection performance. In this paper we propose an effective representation, Multi-Order Contextual co-Occurrence (MOCO), to implicitly model the high level context using solely detection responses from a baseline object detector. The so-called (1^{st}-order) context feature is computed as a set of randomized binary comparisons on the response map of the baseline object detector. The statistics of the 1^{st}-order binary context features are further calculated to construct a high order co-occurrence descriptor. Combining the MOCO feature with the original image feature, we can evolve the baseline object detector to a stronger context aware detector. With the updated detector, we can continue the evolution till the contextual improvements saturate. Using the successful deformable-part-model detector [13] as the baseline detector, we test the proposed MOCO evolution framework on the PASCAL VOC 2007 dataset [8] and Caltech pedestrian dataset [7]: The proposed MOCO detector outperforms all known state-of-the-art approaches, contextually boosting deformable part models (ver.5) [13] by 3.3% in mean average precision on the PASCAL 2007 dataset. For the Caltech pedestrian dataset, our method further reduces the log-average miss rate from 48% to 46% and the miss rate at 1 FPPI from 25% to 23%, compared with the best prior art [6].

1. Introduction

Detecting objects from static images is an important and yet highly challenging task and has attracted many interests of computer vision researchers in the recent decades [35, 36, 10, 13, 31, 26, 19]. The difficulties originate from various aspects including large intra-class appearance variation, objects deformation, perspective distortion and alignment issues caused by view point change, and the categorical inconsistency between visual similarity and functionality.

According to the recent results of the standards-making PASCAL grand challenge [8], The detection approach based on sliding window classifiers are presently the predominant method. Such methods extract image features in each scan window and classify the features to determine the confidence of the presence of the target object [25, 32, 16]. They are further enriched to incorporate sub-part models of the target objects and the confidences on sub-parts are assembled to improve detection of the whole objects [21, 10].

One key disadvantage of the approaches above is that only the information inside each local scanning window is used: joint information between scanning windows or information out of the scanning window are either thrown away or heuristically exploited through post-processing proce-

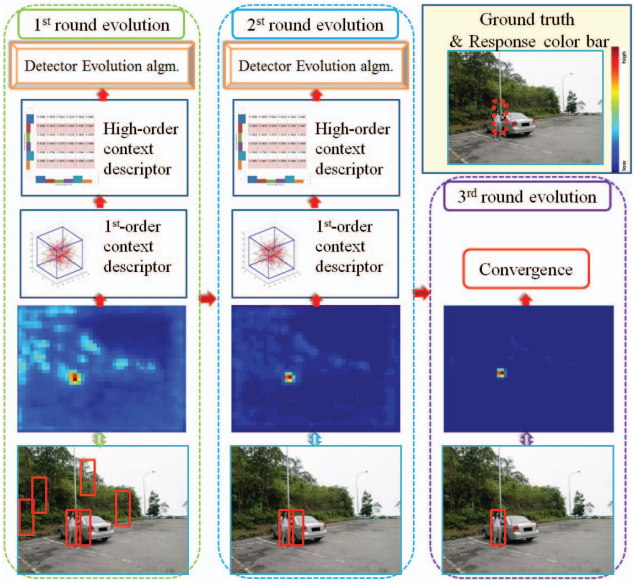

Figure 1: **The proposed MOCO Detection Evolution.** The input image with ground truth label (red dotted rectangle) is shown at top-right corner. The framework evolves the detector using high-order context till the convergence. At each iteration, response map and 0^{th}-order context is computed using the initial baseline detector (for the 1^{st} iteration) or the evolved detector from the prior iteration (for later iterations). Then the 0^{th}-order context is used for computing the 1^{st}-order context, upon which high order co-occurrence descriptors are computed. Finally context in all orders are combined to train a evolving detector. The iteration stops when the overall performance converges. The evolution eliminates many false positives using implicit contextual information, and fortifies the true detections.

dures such as non-maximum suppression. Naturally, to improve detection accuracy, context in the neighborhood of each scan window can provide rich information and should be explored. For example, a scanning window in a pathway region is more likely to be a true detection of human than the one inside a water region. In fact, there have been some efforts on utilizing contextual information for object detection and a variety of valuable approaches have been proposed [14, 27, 28]. High level image contexts such as semantic context [4], image statistics [27], and 3D geometric context [15], are used as well as low level image contexts, including local pixel context [5] and shape context [23].

Besides utilizing context information from the original image directly, another line of works including Spatial Boost [1], Auto-Context [29], and the extensions elegantly integrate the classifier responses from nearby background pixels to help determine the target pixels of interest. These works have been applied successfully to solve problems such as image segmentation and body pose estimation. Inspired by these prior arts, Contextual Boost [6] was proposed to extract multi-scale contextual cues from the detector response map to boost the detection performance. Contextual information directly from the responses of multiple object detectors has also been explored. In [18, 20, 34] the co-occurrence information among different object categories is extracted to improve the performance in various classification tasks. Such methods require multiple base object classifiers and generally necessitate a fusion classifier to incorporate the co-occurrence information, making them expensive and sensitive to the performance of individual base classifiers.

In this paper we aim at developing an effective and generic approach to utilize contextual information without resorting to the multiple object detectors. The rationale is that, even though there is only one classifier/detector, higher order contextual information such as the co-occurrence of objects of different categories can still be implicitly and effectively used by carefully organizing the responses from a single object detector. Since only one classifier is available, the co-occurrence of different object types cannot be explicitly encoded as the multi-class approaches. However, the difference among the responses of the single classifier on different object regions implicitly conveys such contextual information. An example is illustrated in Fig.(1). The responses of a pedestrian detector to various object regions such as the sky, streets, and trees, may vary greatly, but a homogeneous region of the response map corresponds to a region with semantic similarity. Actually, the initial response map in Fig.(1) can lead to a rough tree, sky and street segmentation. This reasoning hints a possibility to encode higher order contextual information with single object detection response. Therefore, if we treat the single classifier response map as an "image", we can extract descriptors to represent high order contextual information.

Our multi-order context representation is inspired by the recent success of randomized binary image descriptors [22, 3, 24]. First we propose a series of binary features where each bit encodes the relationship of classification response values for a pair of pixels. The difference of detector responses at different pixels implicitly captures the contextual co-occurrence patterns pertinent to detection improvements. Recent research also shows that image patches could be more effectively classified with higher-order co-occurrence features [17]. Accordingly we further propose a novel high order contextual descriptor based on the binary pattern of comparisons. Our high order contextual descriptor captures the co-occurrence of binary contextual features based on their statistics in the local neighborhood. The context features at all different orders are complementary to each other and are therefore combined together to form a multi-order context representation.

Finally the proposed multi-order context representations are integrated into an iterative classification framework, where the classifier response map from the previous iteration is further explored to supply more contextual constraints for the current iteration. This process is a straightforward extension of our contextual boost algorithm in [6]. Similar to [6], since the multi-order contextual feature encodes the contextual relationships between neighborhood image regions, through iterations it naturally evolves to cover greater neighborhoods and incorporates more global contextual information into the classification process. As a result our framework effectively enables the detector evolving to be stronger across iterations. We showcase our "detector evolution" framework using the successful deformable part models [13] as our initial baseline detector. Extensive experiments confirm that our framework achieves better accuracy monotonically through iterations. The number of iterations is determined in the training stage when the detection accuracy converges. On the PASCAL VOC 2007 datasets [8], our method outperforms all state-of-the-art approaches, and improves by 3.3% over the deformable part models (ver.5) [13] in mean average precision. On the Caltech dataset [7], compared with the best prior art achieved by contextual boost [6], our method further reduces the log-average miss rate from 48% to 46% and the miss rate at 1 FPPI from 25% to 23%.

2. Multi-order Context Representation

Fig.(2) summarizes the flow chart for constructing the multi-order context representation from an image. First, the image is densely scanned with sliding windows in a pyramid of different scales. For each location of scan window, image features are extracted and a pre-trained classifier is applied to compute the detection response. The detection response maps for each scale are smoothed as in Sec. 2.1.

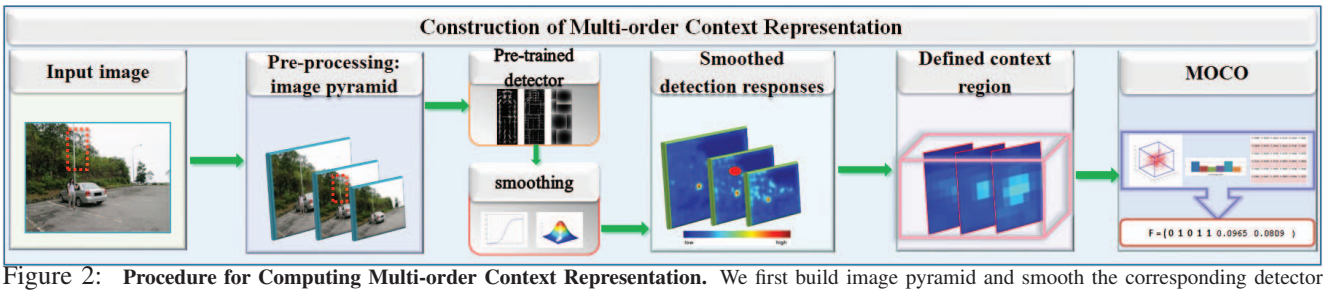

Figure 2: **Procedure for Computing Multi-order Context Representation.** We first build image pyramid and smooth the corresponding detector response map as discussed in Sec. 2.1. For each detection candidate (red dotted rectangle), we locate its position (red dotted rectangle) in the image pyramid and its position (red solid area) in the smoothed detection responses map. We define its context structure $\Omega(p)$ (0^{th}-order) as in Sec. 2.1. Finally we compute the 1^{st}-order binary comparison based context features, upon which we further extract high order co-occurrence descriptor detailed in Sec. 2.3.They are combined as the proposed MOCO descriptors.

We define the context region in terms of spatial and scale for each candidate location. We then compute a series of binary features using randomized comparison of detector responses within the context region, as detailed in Sec. 2.2. Finally, we compute the statistics of the binary comparison features and extract high order co-occurrence descriptors as shown in Sec. 2.3. They together construct the proposed Multi-Order Contextual co-Occurrence (MOCO).

2.1. Context Basis (0^{th}-order)

Intuitively, the appearance of the original image patch containing the neighborhood of target objects provides important contextual cues. However it is difficult to model this kind of context in original image because the neighborhood around target objects may vary dramatically in different scenarios [19]. A logical approach to this problem is: firstly convolve the original image with a particular filter to reduce the diversity of the neighborhood of a true target object as foreground with various backgrounds; then extract context feature from the filtered image. For object detection tasks, we prefer such a filter to be detector driven. Given the observation from Fig.(1) that the positive responses cluster densely around humans but occur sparsely in the background, we simply take the object detector as this specific filter and directly extract context information from the classification response map, denoted as $\mathcal{M}$.

Since the value range of the classification response is $[-\infty, +\infty]$, we first adopt logistic regression to map the value at each pixel s into a grayscale value $s^{'} \in [0, 255]$.

$$s^{'} = \frac{255}{1 + exp(\alpha \cdot s + \beta)}, \tag{1}$$

where $\alpha = -1.5$, $\beta = -\frac{\eta}{\alpha}$, and η is the pre-defined classifier threshold. Eq. (1) turns the response map into a "standard" image, denoted as $\mathcal{M}^{'}$.

The detection responses are usually noisy. To construct context feature from $\mathcal{M}^{'}$, Gaussian smoothing with kernel size 7*7 and std value 1.5 is performed to reduce noise sensitivity, as shown in Fig(1, 2). In the smoothed $\mathcal{M}^{'}$, each pixel $\dot{P}$ represents a local scan window in the original image and its intensity value indicates the detection confidence in the window. Such a response image thus conveys context information, which we denote as 0^{th}-order context.

We define a 3D lattice structure centered at $\dot{P}$ in spatial and scale space. We set $\dot{P}$ as the origin of the local 3-dimensional coordinate system, and index each pixel $\mathbf{a}$ by a 4-dimension vector $[x, y, l, s]$. Here $[x, y]$ refers to the relative location with respect to $\dot{P}$; l represents the relative scale level with respect to $\dot{P}$; s means the value of the pixel $\mathbf{a}$ in the smoothed response image $\mathcal{M}'$, *e.g.* $[2, 3, 2, 175]$ means the pixel $\mathbf{a}$ locates in the 2-level higher than $\dot{P}$, $(2, 3)$ in (x, y)-dimensions relative to $\dot{P}$, with pixel value 175. The context structure $\Omega(\dot{P})$ around $\dot{P}$ in the spatial and scale space is defined as:

$$\mathbf{\Omega}(\dot{P}; W, H, L) = \left\{ (x, y, l, s) \;\middle|\; \begin{array}{l} |x| \leq W/2 \\ |y| \leq H/2 \\ |l| \leq L/2 \end{array} \right\}, \tag{2}$$

where (W, H, L) determines the size and shape of $\Omega(\dot{P})$. For example, $(1, 1, 1)$ means the context structure is a $3 \times 3 \times 3$ cubic region.

2.2. Binary Pattern of Comparisons (1^{st}-order)

Given the 0^{th}-order context structure, we propose to use comparison based binary features to incorporate the co-occurrence of different objects. Although we only have a single object detector, the response values at different locations indicate the confidences of the target object existing. Therefore, each binary comparison encodes the contextual information of whether one location is more likely to contain the target object than the other.

2.2.1 Comparison of Response Values

Specifically, we define the binary comparison τ in the 0^{th}-order context structure $\Omega(\dot{P})$ of size $W \times H \times L$ as:

$$\tau(\mathbf{s}; \mathbf{a}, \mathbf{b}) := \begin{cases} 1 & if \;\; \mathbf{s}(\mathbf{a}) < \mathbf{s}(\mathbf{b}) \\ 0 & otherwise \end{cases}, \tag{3}$$

where $\mathbf{s}(\mathbf{a})$ represents the pixel value in $\Omega(\dot{P})$ at $\mathbf{a} = [\mathbf{x_a}, \mathbf{y_a}, \mathbf{l_a}]$. Naturally selecting a set of n $(\mathbf{a}, \mathbf{b})$-location pairs inside $\Omega(\dot{P})$ uniquely defines a set of binary comparisons. Similar to [3], we define the n-dimensional binary

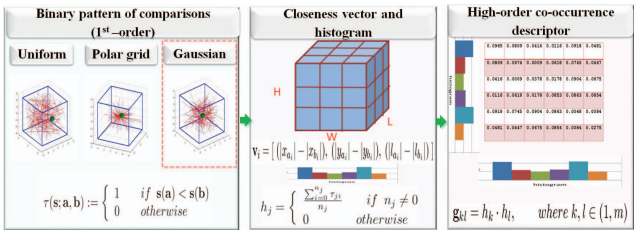

Figure 3: **Multi-order Context Representation.** In the context structure $\Omega(\dot{P})$ with size $W \times H \times L$ around a position $\dot{P}$ (green dot), we first define binary pattern of randomized comparisons (1^{st}-order) based on certain distributions shown on left, described in Sec. 2.2.1 and 2.2.2. We then define the closeness measure $\mathbf{v}_i$ and divide each dimension into t intervals yielding $m = t^3$ subregions (bounded by the solid and dotted red lines), upon which we compute the histogram h_j using Eq. (5,4) as the high-order co-occurrence descriptor.

descriptors $\mathbf{f_n} = [\tau_1, \tau_2, \ldots, \tau_n]$ as our 1^{st}-order context descriptor. However, care needs to be taken for selecting the n specific pairs for the descriptor.

2.2.2 Randomized Arrangement

There are numerous options for selecting n pairs of binary comparisons in Eq. (3). As shown in Fig.(3), two extreme cases of selection are:

(i) The locations of each test pair $(\mathbf{a_i}, \mathbf{b_i})$ are evenly distributed inside $\Omega(\dot{P})$ and binary comparison τ_i can occur far from the origin point: $\mathbf{x_{a_i}}, \mathbf{x_{b_i}} \sim U(-\frac{W}{2}, \frac{W}{2})$,i.i.d; $\mathbf{y_{a_i}}, \mathbf{y_{b_i}} \sim U(-\frac{H}{2}, \frac{H}{2})$,i.i.d; $\mathbf{l_{a_i}}, \mathbf{l_{b_i}} \sim U(-\frac{L}{2}, \frac{L}{2})$,i.i.d;

(ii) The locations of each test pair $(\mathbf{a_i}, \mathbf{b_i})$ concentrate heavily surrounding the origin: $\forall i \in (1, n)$, $\mathbf{a}_i = [0, 0, 0]$, and $\mathbf{b_i}$ lies on any possible position on a coarse 3D polar grid.

Type (i) ignores the facts that the origin of $\Omega(\dot{P})$ represents the location of the detection candidates and thus the context near it might contain more important clues; while type (ii) yields too sparse samples at the boarders of $\Omega(\dot{P})$ to stably capture the complete context information. To address these issues, we adopt a randomized approach:

(iii) $\mathbf{a_i}, \mathbf{b_i} \sim Gaussian(\mu, \Sigma)$, i.i.d. $\mu = [0, 0, 0]$, and $\Sigma = \begin{vmatrix} \epsilon_1 \cdot W^2 & 0 & 0 \\ 0 & \epsilon_2 \cdot H^2 & 0 \\ 0 & 0 & \epsilon_3 \cdot L^2 \end{vmatrix}$. So Σ is correlated with the size of context structure $\Omega(\dot{P})$ and the scaling parameters $[\epsilon_1, \epsilon_2, \epsilon_3]$ are set empirically as $[0.15, 0.15, 0.15]$ that give the best detection rate in our experiments.

The randomized binary features compare the 0^{th}-order context in a set of random patterns and provides rich 1^{st}-order context. The patterns of comparisons capture co-occurrence of classification responses within the context structure $\Omega(\dot{P})$. We can then construct the high order context descriptor using the 1^{st}-order context.

2.3. High Order Co-occurrence Descriptor

It has been shown that higher-order co-occurrence features help improve classification accuracy [17]. Inspired by it, we exploit higher order context information based on the co-occurrence and statistics of the 1^{st}-order context.

Denote $\mathbf{f}_n = [\tau_1, \tau_2, \ldots, \tau_n]$ the randomized co-occurrence binary features, where τ_i corresponds to a comparison between two pixels $\mathbf{a}_i = [x_{a_i}, y_{a_i}, l_{a_i}]$ and $\mathbf{b}_i = [x_{b_i}, y_{b_i}, l_{b_i}]$. For each pair of pixels $\mathbf{a}_i$ and $\mathbf{b}_i$, we define a closeness vector $\mathbf{v}_i = [\, |x_{a_i}| - |x_{b_i}|,\ |y_{a_i}| - |y_{b_i}|,\ |l_{a_i}| - |l_{b_i}| \,]$ to measure the absolute difference of the locations of $\mathbf{a}_i$ and $\mathbf{b}_i$ in x-dimension, y-dimension, l-dimension. For example, $|x_{a_i}| - |x_{b_i}| > 0$ implies that in x-dimension, $\mathbf{a}_i$ is closer to the origin $\dot{P}$ than $\mathbf{b}_i$. Thus $\mathbf{v}_i$ measures whether $\mathbf{a}_i$ or $\mathbf{b}_i$ is closer to $\dot{P}$. This is an important measure as it can be easily observed that stronger detection responses occur in regions closer to the true positive locations. Accordingly the distribution of τ_i w.r.t. $\mathbf{v}_i$ contains important context cues. To compute a stable distribution that is robust against noise, we evenly divide each dimension into t intervals yielding $m = t^3$ subregions, and compute a histogram $\mathbf{h}_m = [h_1, \ldots, h_m]$, as shown in Fig.(3).

Specifically, suppose n_j co-occurrence tests fall into the j-th subregion and their values are $\{\tau_{j_1}, \tau_{j_2}, \ldots, \tau_{j_{n_j}}\}$, the corresponding histogram value h_j is calculated as

$$h_j = \begin{cases} \frac{\sum_{i=0}^{n_j} \tau_{j_i}}{n_j} & if \ \ n_j \neq 0 \\ 0 & otherwise \end{cases} \tag{4}$$

The high order co-occurence descriptor is then constructed as follows,

$$\mathbf{f}_p = \{g_{kl} \mid g_{kl} = h_k \cdot h_l, \ \ _{(k,l=1,\ldots,m)}\}, \tag{5}$$

While the 1^{st}-order co-occurrence features $\mathbf{f}_n$ describes the direct pair-wise relationships between neighborhood positions in a local context, the high order co-occurrence features $\mathbf{f}_p$ capture the correlations among such pair-wise relationships in the local context. Complementarily they provide rich context cues and are combined into the Multi-Order Contextual co-Occurrence (MOCO) descriptor, $\mathbf{f}_c = [\mathbf{f}_n, \mathbf{f}_p]$.

3. Detection Evolution

To effectively use the MOCO descriptor for object detection, we propose an iterative framework that allows the detector to evolve and achieve better accuracy. Such a concept of detection "evolution" had been successfully used for pedestrian detection in Contextual Boost [6]. In this paper, we straightforwardly extend the MOCO based evolution framework to integrate with deformable-part models [10, 13] for general object detection tasks.

3.1. Feature Selection

Our detector uses the MOCO descriptor together with the non-context image features extracted in each scan window in the final classification process. The image features can further consist of more than one descriptors that are computed from different perspectives, e.g., the FHOG descriptors for different parts in the deformable-part-model [10, 13]. As a result, the dimension of the combined feature descriptor can be very high, sometimes more than $10,000$ dimensions. Feeding such features to a general classification algorithm can be unnecessarily expensive. Therefore a step of feature selection is employed when constructing the classifiers at each iteration of detection evolution. Many popular feature selection algorithms have been proposed, such as Boosting [11, 12] or Multiple Kernel Learning [31, 30]. Either of them can be used for our purpose. In our experiments boosting [12] is used for feature selection.

3.2. General Evolution Algorithm

The iterative process of the detector evolution framework is similar to Contextual Boost [6]. Given an initial baseline detector, the iteration procedure for training a new evolving detector is as follows. First, the baseline detector is used to calculate the response maps. Then, the MOCO as well as the image features are extracted on all the training samples. Bootstrapping is used to iteratively add hard samples to avoid over-fitting. Next, feature selection is applied to select the most meaningful features amongst the MOCO and image features. Finally, the selected features are fed into a general classification algorithm to construct a new detector, which will serve as the new baseline detector for the next iteration. As our MOCO is defined in a context region, the iteration will automatically propagate context cues to larger and larger regions. As a result, more and more context will be incorporated through the iterations, and the evolved detectors can yield better performance. The iteration process stops when the performances of the evolving detectors converge. In the testing stage, the same evolution procedure is applied using the learned detectors respectively.

3.3. Integration with Deformable-Part-Model

The deformable-part-model approach [10, 13] has achieved significant success for general object detection tasks. The basic idea is to define a coarse root filter that approximately covers an entire object and higher resolution part filters that cover smaller parts of the object. The relationship between the root and the parts is modeled in a star structure as,

$$s_f = s_r + \sum_{i=1}^{N_p} (s_{p_i} - d_i), \tag{6}$$

where s_r is the detection score of the root filter, s_{p_i} and d_i respectively represent the detection score and deformation cost of the i-th part filter, and N_p is the number of part filters. The star-structural constraints and the final detection are achieved using a latent-SVM model.

From the viewpoint of context, the deformable-part-model essentially exploits the intra context inside the object region, e.g., various arrangements of different parts. In contrast, the proposed MOCO deals with the co-occurrence of scanning windows that cover the object region and its neighborhood. Therefore it exploits the inter context around the object region. Clearly these two kinds of context are exclusive and complementary to each other. This encourages us to combine them together to provide more comprehensive contextual constraints.

Note that Eq. (6) consists of both the final detection response s_f and the detection responses s_{p_i} from the N_p part filters. Since each response s corresponds to a response map, we calculate the MOCO descriptors using each of the response maps. We follow the same procedure of computing the MOCO descriptors $\mathbf{f}_c$ for the root filter from s_f, to obtain the MOCO descriptors $\mathbf{f'}_{c_i}$ for parts on s_{p_i} . Furthermore, to effectively evolve the baseline deformable-part-model detector using the calculated MOCO, we apply the iterative framework not only on the root filter but also on part filters and detectors for every component. The detailed training procedure for integrating our MOCO and the deformable-part-model is summarized in Algm. (1). The input to the algorithm includes the training dataset S_{train} and the deformable-part-model Ψ_0 as the initial baseline detector. In each iteration, we first adopt the same iteration process as in Sec. 3.2 for part filters and the model for each component, and evolve the component model accordingly for the next iteration. This step is shown as step 2 in Algm. (1). Then we use the latent-SVM to fuse the N_c components and retrain an evolved detector for the next iteration. Bootstrapping is again used to avoid over-fitting. The iteration process stops when we observe that the detection accuracy rate converges.

4. Experiments and Discussion

We have conducted extensive experiments to evaluate the proposed MOCO and the detection evolution framework. To demonstrate the advantage of our approach, we adopt the challenging PASCAL VOC 2007 dataset [8] with 20 categories of objects, which are widely acknowledged as one of the most difficult benchmark datasets for general object detection. We use the deformable-part-model [13] with default setting (3 components, each with 1 root and 8 part filters) as our initial baseline detector. First, to demonstrate the advantage of the MOCO, we compare the performance achieved by using different orders of context information. We show performances with various parameter settings to

Algorithm 1: Detection Evolution

Input: Pre-trained deformable-part-model Ψ_0 with N_c components, each containing N_p part filters; training data set $\mathbf{S}_{train}$; detection accuracy rate (*e.g.* average precision) δ_0 of Ψ_0 on $\mathbf{S}_{train}$; convergence threshold ξ.

Output: Iteratively evolved detectors $\Psi_1, \ldots, \Psi_{N_d}$

Set $R = 0$

Do

1. $R = R + 1$, $N_d = R$.
2. **for** $i = 1 \rightarrow N_c$ **do**
   1). Extract the image feature $\mathbf{f}_I$ according to the i_{th} component of $\Psi_{(R-1)}$ on $\mathbf{S}_{train}$.
   2). Compute the detector response maps on S_{train} using $\Psi_{(R-1)}$.
   3). For each detection candidate $\dot{P}$, compute the 1^{st}-order and high-order context descriptors on $\Omega(\dot{P})$ according to Eq. (3, 4, 5) for each of the N_p part filter responses, resulting multiple MOCOs as $[\mathbf{f}_c, \mathbf{f'}_{c_1}, \ldots, \mathbf{f'}_{c_{N_p}}]$
   4). Do feature selection using Boosting [12] on $[\mathbf{f}_I, \mathbf{f}_c, \mathbf{f'}_{c_1}, \ldots, \mathbf{f'}_{c_{N_p}}]$, to learn the informative features $\mathbf{f}_{L_i}$ for the i_{th} component.
   5). Bootstrap and retrain the evolved detector for the i_{th} component.
3. Bootstrap and retrain the evolved detector Ψ_R via latent-SVM [10, 13] for fusing the responses from the N_c evolved component detectors.
4. Evaluate the detection rate δ_R on $\mathbf{S}_{train}$ using Ψ_R.

While $\delta_R - \delta_{(R-1)} > \xi$;

demonstrate the characteristics of the MOCO. Second, we compare the performance at different iterations as the detector evolves to show that the detectors quickly converge in about 2∼3 iterations. Third, we compare the performance of our method with those of state-of-the-art approaches and show substantial improvement. Furthermore, we also experiment on Caltech pedestrian dataset [7], which was used as the main evaluation benchmark for Contextual Boost [6]. The comparisons demonstrate the advantages of our approach.

4.1. Multi-order Context Representation

We first evaluate the MOCO representation and experiment with different parameters settings. We use 5 categories (*plane, bottle, bus, person, tv*) from PASCAL VOC 2007 and experiment on "train" and "val" set for various parameters. All experiments in this section only run 1-iteration of detection evolution. We compare the mean Average Precisions (mAP) to show how the performance varies with different parameter settings.

Context Parameters. Two important parameters that directly affect the computation of context descriptors are the size of Ω_p and the number n of binary comparisons. Since

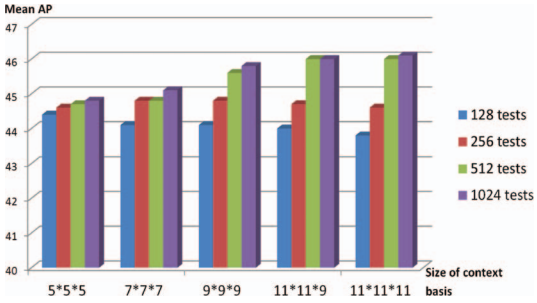

Figure 4: Mean AP (mAP) Varies for Different Parameters: the size $W \times H \times L$ of context structure $\Omega(\dot{P})$ and the number n of binary comparison tests. Only 1^{st}-order context feature and the image features is used for evaluation.

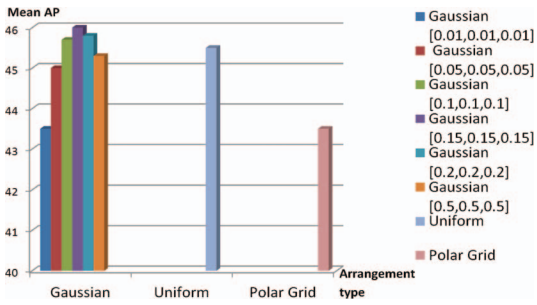

Figure 5: Mean AP (mAP) Varies for Different Arrangements. Only 1^{st}-order context features and the image features is used for evaluation.

the binary comparisons $\{\tau_1, \tau_2, \ldots, \tau_n\}$ are randomly sampled inside the 3D context structure $\Omega(\dot{P})$, the comparison number n is chosen proportional to the size of $\Omega(\dot{P})$, $W \times H \times L$. As shown in Fig.(4), bigger size of $\Omega(\dot{P})$ and number n correspond to richer context information and thus yield better performance, yet requiring more computation. To balance the performance and computational cost, we finally choose $11 \times 11 \times 9$ as $\Omega(\dot{P})$ size, and 512 as the binary comparison test number, where the scale factor is $2^{0.1}$ as in [10] and 9 scales up is about 2 times.

1^{st}-order Context. According to the analysis in Sec. 2.2.2, we choose type iii of Gaussian sampling for constructing the 1^{st}-order context descriptor. We compared the detection performances using different Gaussian parameters. As shown in Fig.(5), the best accuracy is achieved when the variances in the three dimensions are $[0.15, 0.15, 0.15]$ respectively. Fig.(5) also shows the comparison with the sampling methods of type i and type ii, which confirms the advantage of Gaussian sampling.

High Order Context. The most important parameter for computing high order context descriptor is the dimension m of the histogram. Since the high order context descriptor $\mathbf{f}_p$ is complementary to the 1^{st}-order context feature $\mathbf{f}_n$, they are combined when evaluating the detection performance. Table.(1) shows the detection accuracy when choosing different values of m, where the best accuracy is

$m=0$	$m=8$	$m=27$	$m=64$	$m=125$
46.0	46.3	**46.7**	46.5	46.1

Table 1: Mean AP (mAP) varies with respect to the length of high-order co-occurrence feature $\mathbf{f}_p$. The high order context descriptor together with 1^{st}-order context feature and the image features are used. $m=0$ refers to not using any high order feature.

0^{th}	1^{st}	$1^{st}+H$	$0^{th}+1^{st}$	$0^{th}+1^{st}+H$	SURF	LBP
45.5	46.0	46.7	46.8	**47.2**	44.7	45

Table 2: Mean AP (mAP) varies with the combination of different order context feature, where 0^{th}, 1^{st}, H respectively refers to 0^{th}, 1^{st} and high order descriptors. We also compared with SURF [2] or LBP [33] extracted on each level of context structure $\Omega(\dot{P})$.

0	1	2	3(converged)	4	5	6
35.4	37.6	38.3	38.7	38.8	38.7	38.7

Table 3: Mean AP (mAP) varies with respect to the proposed detection evolution algorithm, where 0-iteration in the left refers to the baseline without detection evolution.

achieved when the closeness vector space is divided into $m=27(=3^3)$ subregions.

Context in Different Orders. To show that different orders of context provide complimentary constraints for object detection, we compared the detection accuracy using different combinations of the multi-order context descriptors. For 0^{th}-order context, we chose the best parameter settings presented in [6]. As shown in Table.(2), clearly the MOCO descriptor that combines all orders of context achieves the best detection performance. This confirms that none of the multi-order contexts is redundant. Another way of exploring the 1^{st}-order context is to extract the gradient-based features such as SURF [2] or LBP [33] directly on each scale of the context structure $\Omega(\dot{P})$. However it does not help improve the accuracy in our experiments, as shown in Table.(2). This means that the context across larger spatial neighborhood or different scales can be more effective than the context conveyed by local gradients between adjacent positions.

4.2. Detector Evolution

Using the best parameters for the MOCO descriptor obtained using the "train" and "val" datasets, we evaluate the detector evolution process across iterations. The entire PASCAL dataset is used as the testbed, e.g., training on "trainval" and testing on "test" [8]. We run Algm. (1) and compare the detection accuracy through iterations. For most categories, our framework converges at the second or third iteration. To better show the trend of the detector evolution process, we keep it running for 6 iterations. As shown in Table.(3), the accuracy is steadily improved through iterations and converges quickly.

4.3. Comparison with State of Art

Finally, we compare the overall performance of our approach with the state of art.

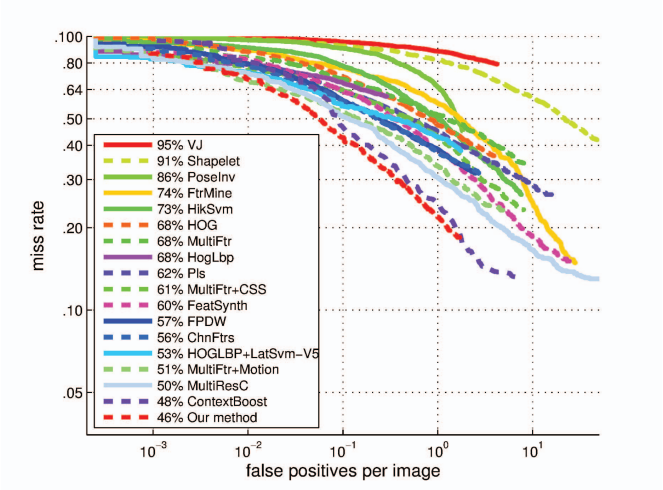

Figure 6: The comparison between our algorithm and the state of the arts in Caltech Pedestrian test dataset.

PASCAL VOC 2007. We first compare our method with state-of-the-art approaches on PASCAL dataset [8]. As shown in Table.(4), our algorithm stably outperforms the baselines [13] in all 20 categories. Especially on the categories of sheep, tv, and monitor, the algorithm achieves significant AP improvements by $6.6\%, 5.7\%$. When compared with all prior arts, our approach outperforms 12 out of 20 categories, and achieves the highest mean AP (mAP) at 38.7, outperforming the deformable model (ver.5) [13] by 3.3%.

Caltech Pedestrian Dataset. We also experiment our algorithm on Caltech pedestrian dataset [7]. We follow the same experimental setup as [6, 7] for evaluations. We use LBP [33] to capture the texture information and FHOG [10] to describe the shape information, and only consider "reasonable" pedestrians of 50 pixels or taller with no occlusion or part occlusion [6, 7]. We compare our algorithm with the state-of-the-art results surveyed in [7], as shown in Fig.(6): the best reported log-average miss rate is 48% [6], while our algorithm further lowers the miss rate to 46%. If we consider the miss rate at 1 FPPI, the best reported result is 25% [6], and our algorithm achieves 23%.

4.4. Processing Speed

Our detection evolution framework needs to evaluate each test image N_d times, where N_d is the number of evolved detectors. The experiments show that it generally converges after 2 or 3 iterations and thus the computational cost would be around 2 or 3 times of the deformable part models (ver.5) [13]. On PASCAL dataset [8], for a 500×375 images, it takes about 12 seconds. One way to speed up the detection is to adopt the cascade scheme. In that case most negative candidates can be rejected in early cascades, and the detection could be around 10 times faster [9].

	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	**mAP**
Leo [36]	29.4	55.8	9.4	14.3	28.6	44.0	51.3	21.3	20.0	19.3	25.2	12.5	50.4	38.4	36.6	15.1	19.7	25.1	36.8	39.3	29.6
CMO [19]	31.5	61.8	12.4	18.1	27.7	51.5	59.8	24.8	**23.7**	27.2	30.7	13.7	60.5	51.1	43.6	14.2	19.6	38.5	49.1	44.3	35.2
Det-Cls [26]	38.6	58.7	**18.0**	18.7	31.8	53.6	56.0	**30.6**	23.5	31.1	**36.6**	20.9	62.6	47.9	41.2	**18.8**	23.5	41.8	**53.6**	45.3	37.7
Oxford [31]	37.6	47.8	15.3	15.3	21.9	50.7	50.6	30.0	17.3	**33.0**	22.5	**21.5**	51.2	45.5	23.3	12.4	23.9	28.5	45.3	48.5	32.1
NLPR [35]	36.7	59.8	11.8	17.5	26.3	49.8	58.2	24.0	22.9	27.0	24.3	15.2	58.2	49.2	44.6	13.5	21.4	34.9	47.5	42.3	34.3
Ver.5 [13]	36.6	62.2	12.1	17.6	28.7	54.6	60.4	25.5	21.1	25.6	26.6	14.6	60.9	50.7	44.7	14.3	21.5	38.2	49.3	43.6	35.4
Our method	**41.0**	**64.3**	15.1	**19.5**	**33.0**	**57.9**	**63.2**	27.8	23.2	28.2	29.1	16.9	**63.7**	**53.8**	**47.1**	18.3	**28.1**	**42.2**	53.1	**49.3**	**38.7**

Table 4: Comparison with the state-of-the-art performance of object detection on PASCAL VOC 2007 (trainval/test).

5. Conclusion

In this paper we have proposed a novel multi-order context representation that effectively exploits co-occurrence contexts of different objects, denoted as MOCO, even though we only use detectors for a single object. We preprocess the detector response map and extract the 1^{st}-order context features based on randomized binary comparison and further develop a high order co-occurrence descriptor based on the 1^{st}-order context. Together they form our MOCO descriptor and are integrated into a "detection evolution" framework as a straightforward extension of Contextual Boost [6]. Furthermore, we have proposed to combine our multi-order context representation with the recently proposed deformable part models [13] to supply a comprehensive coverage over both inter-contexts among objects and inner-context inside the target object region. The advantages of our approach are confirmed by extensive experiments. As the future work, we plan to further extend our MOCO to temporal context from videos and contexts from multiple object detectors or multi-class problems.

Acknowledgement

This work was done during the internship of the first author at Epson Research and Development Inc. in San Jose, CA.

References

[1] S. Avidan. SpatialBoost: adding spatial reasoning to adaboost. In *ECCV*, 2006. 2

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 2008. 7

[3] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. In *ECCV*, 2010. 2, 3

[4] P. Carbonetto, N. de Freitas, and K. Barnard. A statistical model for general contextual object recognition. In *ECCV*, 2004. 2

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2

[6] Y. Ding and J. Xiao. Contextual boost for pedestrian detection. In *CVPR*, 2012. 1, 2, 4, 5, 6, 7, 8

[7] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *PAMI*, 2011. 1, 2, 6, 7

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1, 2, 5, 7

[9] P. F. Felzenszwalb, R. B. Girshick, and D. Mcallester. Cascade object detection with deformable part models. In *CVPR*, 2010. 7

[10] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 2010. 1, 4, 5, 6, 7

[11] Y. Freund. An adaptive version of the boost by majority algorithm. *Machine Learning*, 2001. 5

[12] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 2000. 5, 6

[13] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. Discriminatively trained deformable part models, release 5. http://people.cs.uchicago.edu/ rbg/latent-release5/. 1, 2, 4, 5, 6, 7, 8

[14] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *ECCV*, 2008. 2

[15] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. *IJCV*, 2008. 2

[16] M. Jones, P. Viola, P. Viola, M. J. Jones, D. Snow, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, 2003. 1

[17] T. Kobayashi. Higher-order co-occurrence features based on discriminative co-clusters for image classification. In *BMVC*, 2012. 2, 4

[18] T. Kobayashi and N. Otsu. Bag of hierarchical co-occurrence features for image classification. In *ICPR*, 2010. 2

[19] C. Li, D. Parikh, and T. Chen. Extracting adaptive contextual cues from unlabeled regions. In *ICCV*, 2011. 1, 3, 8

[20] H. Ling and S. Soatto. Proximity distribution kernels for geometric context in category recognition. In *ICCV*, 2007. 2

[21] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV*, 2004. 1

[22] M. Özuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *PAMI*, 2010. 2

[23] D. Ramanan. Using segmentation to verify object hypotheses. In *CVPR*, 2007. 2

[24] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, 2011. 2

[25] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *CVPR*, 2000. 1

[26] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan. Contextualizing object detection and classification. In *CVPR*, 2011. 1, 8

[27] A. Torralba. Contextual priming for object detection. *IJCV*, 2003. 2

[28] A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. In *NIPS*, 2004. 2

[29] Z. Tu and X. Bai. Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *PAMI*, 2010. 2

[30] M. Varma and B. R. Babu. More generality in efficient multiple kernel learning. In *ICML*, 2009. 5

[31] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *ICCV*, 2009. 1, 5, 8

[32] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 2004. 1

[33] X. Wang, X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *ICCV*, 2009. 7

[34] Y. Yang and S. Newsam. Spatial pyramid co-occurrence for image classification. In *ICCV*, 2011. 2

[35] J. Zhang, K. Huang, Y. Yu, and T. Tan. Boosted local structured hog-lbp for object localization. In *CVPR*, 2010. 1, 8

[36] L. Zhu, Y. Chen, A. L. Yuille, and W. T. Freeman. Latent hierarchical structural learning for object detection. In *CVPR*, 2010. 1, 8