Learning Random Attributed Relational Graph for Part-based Object Detection

Columbia University ADVENT Technical Report #212-2005-6, May 2005

Dong-Qing Zhang *Shih-Fu Chang*
Department of Electrical Engineering
Columbia University
New York City, NY 10027
Email: {*dqzhang,sfchang*}*@ee.columbia.edu*

Abstract

Part-based object detection methods have been shown intuitive and effective in detecting general object classes. However, their practical power is limited due to the need of part-level labels for supervised learning and the low learning speed. In this report, we present a new model called Random Attributed Relational Graph (RARG), by which we show that part matching and model learning can be achieved by combining variational learning methods with the part-based representations. We also discover an important mathematical property relating the object detection likelihood ratio and the partition functions of the Markov Random Field (MRF) in the model. Our approach demonstrates clear benefits over the state of the art in part-based object detection - 2 to 5 times faster in learning with almost the same detection accuracy. The improved learning efficiency allows us to extend the single RARG model to a mixture model for learning and detecting multi-view objects.

Key Words: Attributed Relational Graph, Random structure, Random graph, Random Attributed Relational Graph, Unsupervised learning, Statistical inference, Graphical model

1 Introduction

The learning-based object detection paradigm recognizes objects in images by learning statistical object models from a corpus of training data. Among many solutions, the part-based approach represents the object model as a collection of parts with constituent attributes and inter-part relationships. Recently, combination of advanced machine learning techniques with the part-based model has shown great promise in accurate detection of a broad class of objects.

Research on statistical part-based models has focused on two fundamental problems: (1) accurate matching between observed parts in the image and the object model and (2) efficient learning of the object model parameters that characterize the statistical distributions of the part attributes and relations. Most prior work in this area focuses on the part-matching problem, namely, finding the correspondence between the detected parts in the image and parts in the object model. For instance, Markov Random Field (MRF)[8][2] formulates the part matching problem as *maximum a posteriori* (MAP) estimation, where both the image and the object model are represented as Attributed Relational Graphs (ARG) [6]. Learning of model parameters requires the ground truth of part correspondences, which are hard to obtain given the large number (15-30) of the parts in a typical image. Another well-known model, called pictorial structure [4], represents an object model as a star-graph and provides an efficient method for locating parts. The method focuses on finding the optimal locations of the parts in the image instead of detecting presence/absence of the object. Similar to the MRF model mentioned above, the main limitation is that learning of model parameters requires the ground truth of the parts correspondences. Different from the MRF model and the pictorial structure, the constellation model developed in [5][10] computes the object-level detection score by estimating the likelihood ratio. The formulation enables the parameters to be learned in an unsupervised manner, i.e. part correspondences need not to be manually labeled. The constellation model represents the inter-part spatial relationships as a joint Gaussian. In order to achieve translation and rotation invariance, the centroid and the orientation of the object has to be estimated and calibrated in the part-matching search algorithm. In addition, the algorithm relies on a state-space search algorithm called A-star to find the optimal part matching without considering other possible correspondences. The lack of consideration for other possible correspondences may degrade the detection accuracy and affect the overall learning efficiency in cases when the single maximal-likelihood correspondence is incorrect. Actually the initial correspondence is very likely to be incorrect when the randomly initialized object model is inaccurate during the initial stage of the learning process. This seems to be confirmed in the experiment results reported in [5]: 40-100 Expectation-Maximization (E-M) iterations and 36-48 hours are required to learn one object class. Finally, for multi-view object classes, the constellation of the parts corresponding to different views cannot be modelled as a global joint distribution.

We propose a novel model, called Random Attributed Relational Graph (RARG), as an extension of the conventional random graph [3]. It is partly inspired by the pictorial structure model and the MRF model described above. We model an object instance as an ARG, with nodes in the ARG representing the parts in the object. In order to explicitly represent the statistics of the part appearance and relations, we associate random variables to the nodes and edges of the graph, resulting in the RARG model. An image containing the object is an instance generated from the RARG plus some patches generated from the background model, resulting in an ARG representation. This graph-based representation makes it easier to handle translation and rotation invariance. And because it represents the part inter-relationship locally by the edges of the RARG rather than a global constellation, the model can be potentially used to model multi-view object classes.

Given the model RARG and the image ARG, we define an Association Graph, each of whose nodes indicates a one-to-one correspondence between one part in the image and one node in the object model. In comparison, the pictorial structure and the MRF model do not provide such interpretation based on statistical generative models. For learning and part matching, we map the parameters of the RARG to a pairwise binary MRF model defined on the Association Graph. We show that there is an elegant mathematical relationship between the object detection likelihood ratio and the partition functions of the MRF. This discovery enables the use of variational inference methods, such as Loopy Belief Propagation or Belief Optimization, to estimate the part matching probability and learn the parameters by variational E-M, and thereby overcomes the low-efficiency problem associated with prior approaches such as the A-star algorithm mentioned above. Finally, our model is able to learn the occlusion statistics of each part through the MRF modelling. In comparison, how to learn the occlusion statistics is not addressed in the constellation model framework[5].

We compare our proposed RARG model with the constellation model developed in [5], which also provides a publicly available benchmark data set . Our approach achieves a significant improvement in learning convergence speed (measured by the number of iteration and the total learning time) with comparable detection accuracy. The learning speed is improved by more than two times if we use a combined scheme of Gibbs Sampling and Belief Optimization, and more than five times if we use Loopy Belief Propagation. The improved efficiency is important in practical applications, as it allows us to rapidly deploy the method to learning general object classes as well as detection of objects with view variations.

We extend the presented RARG model to a Mixture of RARG (MOR) model to capture the structural and appearance variations of the objects with different views in one object class. Through a semi-supervised learning scheme, the MOR model is shown to improve the detection performance against the single RARG model for detecting objects with continuous view variations in a data set consisting of images downloaded from web. The data set, which is constructed by us, can be used for the public benchmark for multi-view object detection.

The report is organized as follows: In section 2.1, The Baysian classification framework is established for the ARG and RARG models. In section 2.2, we describe how to map the RARG parameters to the parameters of Markov Random Field(MRF), and relate the likelihood ratio for object detection to the partition functions of the MRFs. In section 2.3, we present the methods for calculating the partition functions. In section 2.4, the methods for learning RARG are described. Section 2.5 addresses the problem of spatial relational features and provide solutions to solve it. The RARG model is then extended to a mixture model in section 3. Finally, we present the experiments and analysis in section 4.

2 The Random Attributed Relational Graph Model

An object instance or image can be represented as an Attributed Relational Graph [6], formally defined as

Definition 1. *An Attributed Relational Graph(ARG) is a triple $O = (V, E, Y)$, where V is the vertex set, E is the edge set, and Y is the attribute set that contains attribute y_u attached to each node $n_u \in V$, and attribute y_{uv} attached to each edge $e_w = (n_u, n_v) \in E$.*

For an object instance, a node in the ARG corresponds to one part in the object. attributes y_u and y_{uv} represent the appearances of the parts and relations among the parts. For an object model, we use a graph based representation similar to the ARG but attach random variables to the nodes and edges of the graph, formally defined as a Random Attributed Relational Graph

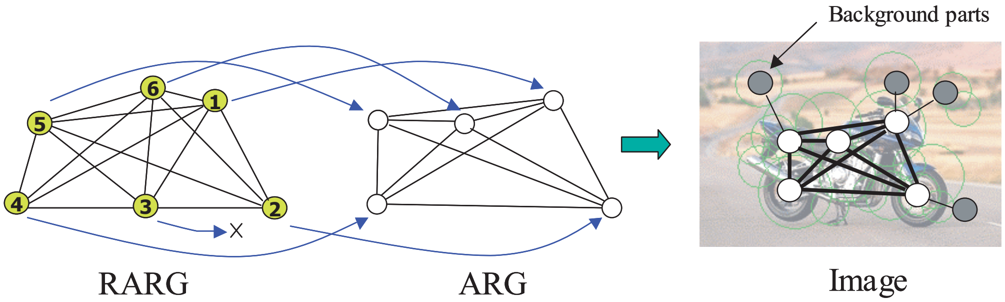


Figure 1: A generative process that generates the part-based representation of an image

Definition 2. *A Random Attributed Relational Graph (RARG) is a quadruple $R = (V, E, A, T)$, where V is the vertex set, E is the edge set, A is a set of random variables consisting of A_i attached to the node $n_i \in V$ with pdf $f_i(.)$, and A_{ij} attached to the edge $e_k = (n_i, n_j) \in E$ with pdf $f_{ij}(.)$. T is a set of binary random variables, with T_i attached to each node (modelling the presense/absence of nodes).*

$f_i(.)$ is used to capture the statistics of the part appearance. $f_{ij}(.)$ is used to capture the statistics of the part relation. T_i is used to model the part occlusion statistics. $r_i = p(T_i = 1)$ is referred to as the *presence probability* of the part i in the object model. An ARG hence can be considered as an instance generated from RARG by multiple steps: first draw samples from $\{T_i\}$ to determine the topology of the ARG, then draw samples from A_i and A_{ij} to obtain the attributes of the ARG and thus the appearance of the object instance. In our current system, both RARG and ARG are fully connected. However, in more general cases, we can also accommodate edge connection variations by attaching binary random variables T_{ij} to the edges, where $T_{ij} = 1$ indicates that there is an edge connecting the node i and node j, $T_{ij} = 0$ otherwise.

2.1 Bayes Classification under RARG Framework

Conventionally, object detection is formulated as a binary classification problem with two hypotheses: $H = 1$ indicates that the image contains the target object (e.g. bike), $H = 0$ otherwise. Let O denote the ARG representation of the input image. Object detection problem therefore is reduced to the following likelihood ratio test

$$\frac{p(O|H=1)}{p(O|H=0)} > \frac{p(H=0)}{p(H=1)} = \lambda \tag{1}$$

Where λ is used to adjust the precision and recall performance. The main problem is thus to compute the positive likelihood $p(O|H = 1)$ and the negative likelihood $p(O|H = 0)$. $p(O|H = 0)$ is the likelihood assuming the image is a *background* image without the target object. Due to the diversity of the *background* images, we adopt a simple decomposable *i.i.d.* model for the background parts. We factorize the negative likelihood as

$$p(O|H=0) = \prod_u p(y_u|H=0) \prod_{uv} p(y_{uv}|H=0) = \prod_u f^-_{B_1}(y_u) \prod_{uv} f^-_{B_2}(y_{uv}) \tag{2}$$

where $f^-_{B_1}(\cdot)$ and $f^-_{B_2}(\cdot)$ are *pdf*s to capture the statistics of the appearance and relations of the parts in the *background* images, referred to as background *pdf*s. The minus superscript indicates that the parameters of the *pdf*s are learned from the negative data set. To compute the positive likelihood $p(O|H = 1)$, we assume

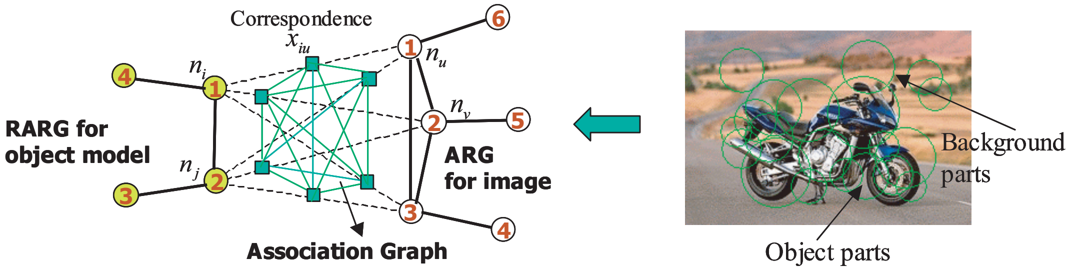


Figure 2: ARG, RARG and the Association Graph. Circles in the image are detected parts

that an image is generated by the following generative process (Figure 1): an ARG is first generated from the RARG, additional patches, whose attributes are sampled from the background *pdf*s, are independently added to form the final part-based representation O of the image. In order to compute the positive likelihood, we further introduce a variable X to denote the correspondences between parts in the ARG O and parts in the RARG R. Treating X as a hidden variable, we have

$$p(O|H=1) = \sum_X p(O|X, H=1)p(X|H=1) \tag{3}$$

Where X consists of a set of binary variables, with $x_{iu} = 1$ if the part i in the object model corresponds to the part u in the image, $x_{iu} = 0$ otherwise. If we assign each x_{iu} a node, then these nodes form an Association Graph as shown in Figure 2. The Association Graph can be used to define an undirected graphical model (Markov Random Field) for computing the positive likelihood in Equation (3). In the rest of the paper, iu therefore is used to denote the index of the nodes in the Association Graph. A notable difference between our method and the previous methods [5][8] is that we use a binary random representation for the part correspondence. Such representation is important as it allows us to prune the MRF by discarding nodes associated with a pair of dissimilar parts to speed up part matching, and readily apply efficient inference techniques such as Belief Optimization[9][11].

2.2 Mapping the RARG parameters to the Association Graph MRF

The factorization in Eq. (3) requires computing two components $p(X|H=1)$ and $p(O|X, H=1)$. This section describes how to map the RARG parameters to these two terms as well as construct MRFs to compute the likelihood ratio.

First, $p(X|H=1)$, the prior probability of the correspondence, is designed so as to satisfy the one-to-one part matching constraint, namely,one part in the object model can only be matched to one part in the image, vice versa. Furthermore, $p(X|H=1)$ is also used to encode the *presence probability* r_i. To achieve these, $p(X|H=1)$ is designed as a binary pairwise MRF with the following Gibbs distribution

$$p(X|H=1) = \frac{1}{Z} \prod_{iu,jv} \psi_{iu,jv}(x_{iu}, x_{jv}) \prod_{iu} \phi_{iu}(x_{iu}) \tag{4}$$

Where Z is the normalization constant, a.k.a the partition function. $\psi_{iu,jv}(x_{iu}, x_{jv})$ is the two-node potential function defined as

$$\psi_{iu,jv}(1,1) = \varepsilon, \quad for \quad i=j \;\; or \;\; u=v; \qquad \psi_{iu,jv}(x_{iu}, x_{jv}) = 1, \;\; otherwise \tag{5}$$

where ε is set to 0 (for Gibbs Sampling) or a small positive number (for Loopy Belief Propagation). Therefore, if the part matching violates one-to-one constraint, the prior probability would drop to zero (or near zero). $\phi_{iu}(x_{iu})$ is the one-node potential function. Adjusting $\phi_{iu}(x_{iu})$ affects the distribution $p(X|H=1)$, therefore it is related to the *presence probability* r_i. By designing $\phi_{iu}(x_{iu})$ to different values, we will result in different r_i. For any iu, we have two parameters to specify for $\phi_{iu}(.)$, namely $\phi_{iu}(1)$ and $\phi_{iu}(0)$. Yet, it is not difficult to show that for any iu, different $\phi_{iu}(1)$ and $\phi_{iu}(0)$ with the same ratio $\phi_{iu}(1)/\phi_{iu}(0)$ would result in the same distribution $p(X|H=1)$ (but different partition function Z). Therefore, we can just let $\phi_{iu}(0)=1$ and $\phi_{iu}(0)=z_i$. Note here that z_i only has the single indice i. meaning the potential function for the correspondence variable between part i in the model and part u in the image does not depend on the index u. Such design is for simplicity and the following relationship between z_i and r_i.

Lemma 1. *r_i and z_i is related by the following equation:*

$$r_i = z_i \frac{\partial \ln Z}{\partial z_i}$$

where Z is the partition function defined in Equation (4).

Proof. To simplify the notations, we assume $N \le M$. It is easy to extend to the case when $N > M$. The partition function can be calculated by enumerating the admissible matching (matching that does not violate the one-to-one constraint) as the following

$$Z(N;M;z_1,z_2,...,z_N) = \sum_X \prod_{iu,jv} \psi_{iu,jv}(x_{iu},x_{jv}) \prod_{iu} \phi_{iu}(x_{iu}) = \sum_{admissible\ X} \prod_{iu} z_i$$

To calculate the above summation, we first enumerate the matchings where there are i nodes $n_{I_1}, n_{I_2}...n_{I_i}$ in the RARG being matched to the nodes in ARG, where $1 \le i \le N$,and $I_1, I_2...I_i$ is the index of the RARG node. The corresponding summation is

$$M(M-1)(M-2)...(M-i+1)z_{I_1}z_{I_2}...z_{I_i} = \binom{M}{i} i! z_{I_1}z_{I_2}...z_{I_i}$$

For all matchings where there are i nodes being matched to RARG, the summation becomes

$$\binom{M}{i} i! \sum_{1 \le I_1 < I_2 < ... < I_i \le N} z_{I_1}z_{I_2}...z_{I_i} = \binom{M}{i} i! \Pi_i(z_1,z_2,...,z_N)$$

Where

$$\Pi_i(z_1,z_2,...,z_N) = \sum_{1 \le I_1 < I_2 < ... < I_i \le N} z_{I_1}z_{I_2}...z_{I_i}$$

is known as *Elementary Symmetric Polynomial*. By enumerating the index i from 0 to N, we get

$$Z(N;M;z_1,z_2,...,z_N) = \sum_{i=0}^{N} \binom{M}{i} i! \Pi_i(z_1,z_2,...,z_N) \tag{6}$$

Likewise, for the *presence probability* r_i, we enumerate all matchings in which the node i in the RARG is matched to a node in the ARG, yielding

$$\begin{aligned} r_i &= \frac{1}{Z} M \sum_{j=0}^{N-1} \binom{M-1}{j} j! z_i \Pi_{j|i}(z_1, z_2, ..., z_N) \\ &= \frac{1}{Z} z_i \sum_{j=0}^{N-1} \binom{M}{j} j! \Pi_{j|i}(z_1, z_2, ..., z_N) \\ &= z_i \frac{1}{Z} \partial Z / \partial z_i = z_i \partial \ln(Z) / \partial z_i \end{aligned}$$

Where, we have used the short-hand $\Pi_{j|i}(z_1, z_2, ..., z_N)$, which is defined as

$$\Pi_{j|i}(z_1, z_2, ..., z_N) = \sum_{1 \le I_1 < I_2 < ... I_p, ... < I_j \le N; I_p \neq i, \forall p \in \{1,2,...,j\}} z_{I_1} z_{I_2} ... z_{I_j}$$

For the pruned MRF, which is the more general case, we can separate the summation into two parts, the summation of the terms containing z_i and the summation of those not

$$Z(N; M; z_1, z_2, ..., z_N) = V_1(z_1, z_2, ..., z_i, ...z_N) + V_2(z_1, z_2, ..., z_{i-1}, z_{i+1} ... z_N)$$

Then the *presence probability* r_i is

$$r_i = \frac{V_1}{Z} = \frac{z_i \frac{\partial Z}{\partial z_i}}{Z} = z_i \frac{\partial \ln Z}{\partial z_i}$$

Where we have used the fact that V_1 and Z is the summation of the monomials in the form of $z_{I_1} z_{I_2} ... z_{I_i}$, which holds the relationship

$$z_{I_1} z_{I_2} ... z_{I_i} = z_{I_k} \frac{\partial}{\partial z_{I_k}} (z_{I_1} z_{I_2} ... z_{I_i}), \qquad \forall I_k \in \{I_1, I_2, ..., I_i\}$$

□

The above lemma leads to a simple formula to learn the *presence probability* r_i (section 2.4). However, lemma 1 still does not provide a closed-form solution for computing z_i given r_i. We resort to an approximate solution, through the following lemma.

Lemma 2. *The log partition function satisfy the inequality*

$$\ln Z \le \sum_{i=1}^{N} \ln(1 + M z_i)$$

and the equality holds when N/M tends to zero (N and M are the numbers of parts in the object model and image respectively). For the pruned MRF, the upper bound is changed to

$$\ln Z \le \sum_{i=1}^{N} \ln(1 + d_i z_i)$$

where d_i is the number of the nodes in the ARG that could possibly correspond to the node i in the RARG after pruning the Association Graph.

Proof. We have obtained the closed-form of the partition function Z in the proof of Lemma 1, therefore it is apparent that Z satisfies the following inequality

$$Z = \sum_{i=0}^{N} M(M-1)...(M-i+1)\Pi_i(z_1, z_2, ..., z_N) \leq \sum_{i=0}^{N} M^i \Pi_i(z_1, z_2, ..., z_N) \tag{7}$$

The equality holds when N/M tends to zero. And we have the following relationships

$$\sum_{i=0}^{N} \Pi_i(z_1, z_2, ..., z_N) = 1 + z_1 + z_2 + ... + z_N + z_1 z_2 + ... + z_{N-1} z_N + ... = \prod_{i=1}^{N} (1 + z_i)$$

and

$$M^i \Pi_i(z_1, z_2, ..., z_N) = \Pi_i(Mz_1, Mz_2, ..., Mz_N)$$

Therefore, the RHS in equation (7) can be simplified as the following

$$\sum_{i=0}^{N} M^i \Pi_i(z_1, z_2, ..., z_N) = \sum_{i=0}^{N} \Pi_i(Mz_1, Mz_2, ..., Mz_N) = \prod_{i=1}^{N} (1 + Mz_i)$$

The above function in fact is the partition function of the Gibbs distribution if we remove the one-to-one constraints. Likewise, for the pruned MRF, the partition function is upper-bounded by the partition function of the Gibbs distribution if we remove the one-to-one constraints, which, by enumerating the matchings, can be written as

$$1 + d_1 z_1 + d_2 z_2 + ... + d_N z_N + d_1 d_2 z_1 z_2 + ... = \prod_{i=1}^{N} (1 + d_i z_i)$$

Therefore we have

$$\ln Z \leq \prod_{i=1}^{N} (1 + d_i z_i)$$

□

Since the closed form solution for mapping r_i to z_i is unavailable, we use the upper bound as an approximation. Consequently, combining lemmas 1 and 2 we can obtain the following relationship for the pruned MRF. $z_i = r_i/((1 - r_i)d_i)$.

The next step is to derive the conditional density $p(O|X, H = 1)$. Assuming that y_u and y_{uv} are independent given the correspondence, we have

$$p(O|X, H = 1) = \prod_{uv} p(y_{uv}|x_{1u}, x_{1v}, ..., x_{Nu}, x_{Nv}, H = 1) \prod_{u} p(y_u|x_{1u}, ..., x_{Nu}, H = 1)$$

Furthermore, y_u and y_{uv} should only depends on the RARG nodes that are matched to u and v. Thus

$$\begin{aligned} p(y_u|x_{11} = 0, ..., x_{iu} = 1, ..., x_{NM} = 0, H = 1) &= f_i(y_u) \\ p(y_{uv}|x_{11} = 0, ..., x_{iu} = 1, x_{jv} = 1, ..., x_{NM} = 0, H = 1) &= f_{ij}(y_{uv}) \end{aligned} \tag{8}$$

Also, if there is no node in the RARG matched to u, then y_u,y_{uv} should be sampled from the background *pdf*s, i.e.

$$\begin{aligned} p(y_u|x_{11} = 0, x_{iu} = 0, ..., x_{NM} = 0, H = 1) &= f^{+}_{B_1}(y_u) \\ p(y_{uv}|x_{11} = 0, x_{iu} = 0, ..., x_{NM} = 0, H = 1) &= f^{+}_{B_2}(y_{uv}) \end{aligned} \tag{9}$$

where $f^+_{B_1}(\cdot)$ and $f^+_{B_2}(\cdot)$ is the background *pdf* trained from the positive data set. Note that here we use two sets of background *pdf*s to capture the difference of the background statistics in the positive data set and that in the negative data set.

Combining all these elements together, we would end up with another MRF (to be described in theorem 1). It is important and interesting to note that the likelihood ratio for object detection is actually related to the partition functions of the MRFs through the following elegant relationship.

Theorem 1. *The likelihood ratio is related to the partition functions of MRFs as the following*

$$\frac{p(O|H=1)}{p(O|H=0)} = \sigma \frac{Z'}{Z} \tag{10}$$

where Z is the partition function of the Gibbs distribution $p(X|H=1)$. Z' is the partition function of the Gibbs distribution of a new MRF, which happens to be the posterior probability of correspondence $p(X|O,H=1)$, with the following form

$$p(X|O,H=1) = \frac{1}{Z'} \prod_{iu,jv} \varsigma_{iu,jv}(x_{iu}, x_{jv}) \prod_{iu} \eta_{iu}(x_{iu}) \tag{11}$$

where the one-node and two-node potential functions have the following forms

$$\eta_{iu}(1) = z_i f_i(y_u)/f^+_{B_1}(y_u); \quad \varsigma_{iu,jv}(1,1) = \psi_{iu,jv}(1,1) f_{ij}(y_{uv})/f^+_{B_2}(y_{uv}) \tag{12}$$

all other values of the potential functions are set to 1 (e.g. $\eta_{iu}(x_{iu}=0)=1$). σ is a correction term

$$\sigma = \prod_u f^+_{B_1}(y_u)/f^-_{B_1}(y_u) \prod_{uv} f^+_{B_2}(y_{uv})/f^-_{B_2}(y_{uv})$$

Proof. We start from the posterior probability $p(X|O,H=1)$. According to the Bayes rule

$$p(X|O,H=1) = \frac{1}{C} p(O|X,H=1) p(X|H=1)$$

where C is the normalization term, which happens to be the positive likelihood $p(O|H=1)$:

$$C = \sum_X p(O|X,H=1) P(X|H=1) = p(O|H=1) \tag{13}$$

Next, let us rewrite the posterior probability $p(X|O,H=1)$ as the following

$$p(X|O,H=1) = \frac{\prod_u f^+_{B_1}(y_u) \prod_{uv} f^+_{B_2}(y_{uv})}{CZ} \frac{p(O|X,H=1)p(X|H=1)Z}{\prod_u f^+_{B_1}(y_u) \prod_{uv} f^+_{B_2}(y_{uv})} \tag{14}$$

Using the independence assumption

$$p(O|X,H=1) = \prod_{uv} p(y_{uv}|x_{1u}, x_{1v}, ..., x_{Nu}, x_{Nv}, H=1) \prod_u p(y_u|x_{1u}, ..., x_{Nu}, H=1)$$

and plugging in the parameter mapping equations in Eq.(8) and (9). Comparing the term in Eq.(14) and the term in the Gibbs distribution in Eq.(11), we note that for any matching X, we have

$$\frac{p(O|X,H=1)p(X|H=1)Z}{\prod_u f^+_{B_1}(y_u) \prod_{uv} f^+_{B_2}(y_{uv})} = \prod_{iu,jv} \varsigma_{iu,jv}(x_{iu}, x_{jv}) \prod_{iu} \eta_{iu}(x_{iu})$$

Furthermore, the posterior probability $p(X|O, H = 1)$ and the Gibbs distribution in Eq.(11) have the same domain. Therefore, the normalization constant should be also equal, i.e.

$$\frac{CZ}{\prod_u f^+_{B_1}(y_u) \prod_{uv} f^+_{B_2}(y_{uv})} = Z'$$

Therefore the positive likelihood is

$$p(O|H = 1) = C = \frac{Z'}{Z} \prod_u f^+_{B_1}(y_u) \prod_{uv} f^+_{B_2} \tag{15}$$

and the likelihood ratio is

$$\frac{p(O|H = 1)}{p(O|H = 0)} = \frac{\prod_u f^+_{B_1}(y_u) \prod_{uv} f^+_{B_2}}{\prod_u f^-_{B_1}(y_u) \prod_{uv} f^-_{B_2}} \frac{Z'}{Z} = \sigma \frac{Z'}{Z} \tag{16}$$

□

2.3 Computing the Partition Functions

Theorem 1 reduces the likelihood ratio calculation to the computation of the partition functions. For the partition function Z, it has a closed form(Eq.(6)) and can be computed in a polynomial time or using the lemma 2 for approximation. The main difficulty is to compute the partition function Z', which involves a summation over all possible correspondences, whose size is exponential in MN. Fortunately, computing the partition function of the MRF has been studied in statistical physics and machine learning [9]. It turns out that, due to its convexity, $\ln Z'$ can be written as a dual function, a.k.a. variational representation, or in the form of the Jensen's inequality [12].

$$\ln Z' \geq \sum_{(iu,jv)} \hat{q}(x_{iu}, x_{jv}) \ln \varsigma_{iu,jv}(x_{iu}, x_{jv}) + \sum_{(iu)} \hat{q}(x_{iu}) \ln \eta_{iu}(x_{iu}) + \mathcal{H}(\hat{q}(X)) \tag{17}$$

Where $\hat{q}(x_{iu})$ and $\hat{q}(x_{iu}, x_{jv})$ are known as one-node and two-node beliefs, which are the approximated marginal of the Gibbs distribution $p(X|O, H = 1)$. $\mathcal{H}(\hat{q}(X))$ is the approximated entropy, which can be approximated by Bethe approximation[12], as below

$$\mathcal{H}(\hat{q}(X)) = -\sum_{iu,jv} \sum_{x_{iu},x_{jv}} \hat{q}(x_{iu}, x_{jv}) \ln \hat{q}(x_{iu}, x_{jv}) + \sum_{iu} (MN - 2) \sum_{x_{iu}} \hat{q}(x_{iu}) \ln(\hat{q}(x_{iu}))$$

Apart from Bethe approximation, it is also possible to use more accurate approximations, such as semidefinite relaxation in [9].

The RHS in the equation (17) serves two purposes, for variational learning and for approximating $\ln Z'$. In both cases, we have to calculate the approximated marginal $\hat{q}(x_{iu})$ and $\hat{q}(x_{iu}, x_{jv})$. There are two options to approximate it, optimization-based approach and Monte Carlo method. The former maximizes the lower bound with respect to the approximated marginal. For example, Loopy Belief Propagation (LBP) is an approach to maximizing the lower bound through fixed point equations[12]. However, we found that, LBP message passing often does not converge using the potential functions in Eq.(5). Nonetheless, we found that if we select the marginal that corresponds to the larger lower bound in Eq.(17) across update iterations, we can achieve satisfactory inference results and reasonably accurate object models.

Another methodology, Monte Carlo sampling[1] , approximates the marginal by drawing samples and summing over the obtained samples. Gibbs Sampling, a type of Monte Carlo method, is used in our system due to its efficiency. In order to reduce the variances of the approximated two-node beliefs, we propose a new method to combine the Gibbs Sampling with the Belief Optimization developed in [11], which proves that there is a closed-form solution (through Bethe approximation) for computing the two-node beliefs given the one-node beliefs and the two-node potential functions (Lemma 1 in [11]). We refer to this approach as Gibbs Sampling plus Belief Optimization (GS+BO).

2.4 Learning Random Attributed Relational Graph

We use Gaussian models for all the *pdf*s associated with the RARG and the background model . Therefore, we need to learn the corresponding Gaussian parameters μ_i,Σ_i,μ_{ij},Σ_{ij};$\mu^+_{B_1}$,$\Sigma^+_{B_1}$,$\mu^+_{B_2}$,$\Sigma^+_{B_2}$;$\mu^-_{B_1}$,$\Sigma^-_{B_1}$,$\mu^-_{B_2}$,$\Sigma^-_{B_2}$. and the *presence probability* r_i.

Learning the RARG is realized by Maximum Likelihood Estimation (MLE). Directly maximizing the positive likelihood with respect to the parameters is intractable, instead we maximize the lower bound of the positive likelihood through Eq.(17), resulting in a method known as Variable Expectation-Maximization (Variational E-M). **Variational E-Step:** Perform GS+BO scheme or Loopy Belief Propagation to obtain the one-node and two-node beliefs.

M-Step: Maximize the overall log-likelihood with respect to the parameters

$$L = \sum_{k=1}^{K} \ln p(O_k|H=1) \tag{18}$$

where K is the number of the positive training instances. Since direct maximization is intractable, we use the lower bound approximation in Eq.(17), resulting in the following equations for computing the parameters

$$\begin{array}{ll}
\xi^k_{iu} = \hat{q}(x^k_{iu}=1), \;\; \xi^k_{iu,jv} = \hat{q}(x^k_{iu}=1, x^k_{jv}=1); & \bar{\xi}^k_{iu} = 1-\xi^k_{iu}, \;\; \bar{\xi}^k_{iu,jv} = 1-\xi^k_{iu,jv} \\
\mu_i = \dfrac{\sum_k \sum_u \xi^k_{iu} y^k_u}{\sum_k \sum_u \xi^k_{iu}} & \Sigma_i = \dfrac{\sum_k \sum_u \xi^k_{iu}(y^k_u-\mu_i)(y^k_u-\mu_i)^T}{\sum_k \sum_u \xi^k_{iu}} \\
\mu_{ij} = \dfrac{\sum_k \sum_{uv} \xi^k_{iu,jv} y^k_{uv}}{\sum_k \sum_{uv} \xi^k_{iu,jv}} & \Sigma_{ij} = \dfrac{\sum_k \sum_{uv} \xi^k_{iu,jv}(y^k_{uv}-\mu_{ij})(y^k_{uv}-\mu_{ij})^T}{\sum_k \sum_{uv} \xi^k_{iu,jv}} \\
\mu^+_{B_1} = \dfrac{\sum_k \sum_u \bar{\xi}^k_{iu} y^k_u}{\sum_k \sum_u \bar{\xi}^k_{iu}} & \Sigma^+_{B_1} = \dfrac{\sum_k \sum_u \bar{\xi}^k_{iu}(y^k_u-\mu_i)(y^k_u-\mu_i)^T}{\sum_k \sum_u \bar{\xi}^k_{iu}} \\
\mu^+_{B_2} = \dfrac{\sum_k \sum_{uv} \bar{\xi}^k_{iu,jv} y^k_{uv}}{\sum_k \sum_{uv} \bar{\xi}^k_{iu,jv}} & \Sigma^+_{B_2} = \dfrac{\sum_k \sum_{uv} \bar{\xi}^k_{iu,jv}(y^k_{uv}-\mu_{ij})(y^k_{uv}-\mu_{ij})^T}{\sum_k \sum_{uv} \bar{\xi}^k_{iu,jv}}
\end{array} \tag{19}$$

The *presence probability* r_i is derived from Lemma 1 using maximum likelihood estimation.Using the lower bound approximation in Eq.(17), we have the approximated overall log-likelihood

$$L \approx \sum_{k=1}^{K} \sum_{iu} \hat{q}(x^k_{iu}=1)\ln z_i - K\ln Z(N;M;z_1,z_2,...,z_N) + \alpha \tag{20}$$

where α is a term independent on the *presence probability* $r_1, r_2, ..., r_N$. To minimize the approximated

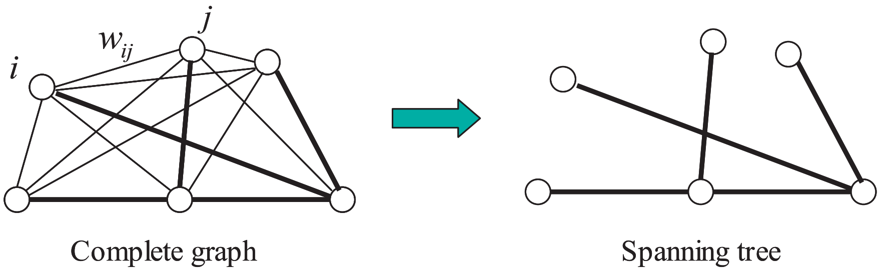


Figure 3: Spanning tree approximation, realized by first constructing a weighted graph (having the same topology as the RARG) with the weight $w_{ij} = |\Sigma_{ij}|$ in the edge $e_l = (n_i, n_j)$, then invoking the conventional minimum spanning tree(MST) algorithm, such as Kruskal's algorithm. Here $|\Sigma_{ij}|$ is the determinant of the covariance matrix of $f_{ij}(.)$ (the *pdf* of the relational feature in the RARG) associated with the edge $e_l = (n_i, n_j)$.

likelihood with respect to z_i, we compute the derivative of the Eq.(20), and equates it to zero

$$\begin{aligned}&\frac{\partial}{\partial z_i}\Big[\sum_{k=1}^{K}\sum_{iu}\hat{q}(x_{iu}^k = 1)\ln z_i\Big] - K\frac{\partial}{\partial z_i}\ln Z(N;M;z_1,z_2,...,z_N)\\ =\ &\sum_{k=1}^{K}\sum_{iu}\hat{q}(x_{iu}^k = 1)\frac{1}{z_i} - K\frac{r_i}{z_i} = 0 \qquad (21)\end{aligned}$$

We used Lemma 1 in the last step. Since $z_i \neq 0$, the above equation leads to the equation for estimating r_i

$$r_i = \frac{1}{K}\sum_{k}\sum_{u}\hat{q}(x_{iu}^k = 1) \qquad (22)$$

For the background parameters $\mu_{B_1}^-$,$\Sigma_{B_1}^-$,$\mu_{B_2}^-$,$\Sigma_{B_2}^-$, the maximum likelihood estimation results in the sample mean and covariance matrix of the part attributes and relations of the images in the negative data set.

2.5 Spanning Tree Approximation for Spatial Relational Features

Our approaches described so far assume the relational features y_{ij} are independent. However, this may not be true in general. For example, if we let y_{ij} be coordinate differences, they are no longer independent. This can be easily seen by considering three edges of a triangle formed by any three parts. The coordinate difference of the third edge is determined by the other two edges. The independence assumption therefore is not accurate. To deal with this problem, we prune the fully-connected RARG into a tree by the spanning tree approximation algorithm, which discards the edges that have high determinant values in their covariance matrix of the Gaussian functions (Figure 3). This assumes that high determinant values of the covariance matrix indicate the spatial relation has large variation and thus is less salient.

In our experiments, we actually found a combined use of fully-connected RARG and the pruned spanning tree is most beneficial in terms of the learning speed and model accuracy. Specifically, we use a two-stage procedure: the fully-connected RARG is used to learn an initial model, which then is used to initialize the model in the 2*nd*-phase iterative learning process based on the pruned tree model. In the detection phase, only spanning-tree approximation is used.

3 Extension to Multi-view Mixture Model

The above described model assumes the training object instances have consistent single views. In order to capture the characteristic of an object class with view variations. We develop a Mixture of RARG (MOR) model, which allows the components in the MOR to capture the characteristic of the objects with different views.

Let R_t denotes the RARG to represent a distinct view t. The object model thereby is represented as $\Re = \{R_t\}$ along with the mixture coefficients $p(R_t|\Re)$. The positive likelihood then becomes

$$p(O|H=1) = \sum_{t=1} p(O|R_t)p(R_t|\Re)$$

The maximum likelihood learning scheme to learn the mixture coefficients and the Guassian *pdf* parameters therefore is similar to that of the Gaussian Mixture Model (GMM), consisting of the following E-M updates

E-step: Compute the assignment probability

$$\zeta_k^t = p(R_t|O_k, \Re) = \frac{p(O_k|R_t)p(R_t|\Re)}{\sum_t p(O_k|R_t)p(R_t|\Re)} \tag{23}$$

M-step: Compute the mixture coefficients

$$p(R_t|\Re) = \frac{1}{N}\sum_k \zeta_k^t \tag{24}$$

and update the Gaussian parameters for each component t(We omit the index t except for ζ_k^t for brevity):

$$\begin{array}{lll}
\xi_{iu}^k = \hat{q}(x_{iu}^k = 1), & \xi_{iu,jv}^k = \hat{q}(x_{iu}^k = 1, x_{jv}^k = 1); & \bar{\xi}_{iu}^k = 1 - \xi_{iu}^k, \ \ \bar{\xi}_{iu,jv}^k = 1 - \xi_{iu,jv}^k \\
\mu_i = \dfrac{\sum_k \zeta_k^t \sum_u \xi_{iu}^k y_u^k}{\sum_k \zeta_k^t \sum_u \xi_{iu}^k}, & \Sigma_i = \dfrac{\sum_k \zeta_k^t \sum_u \xi_{iu}^k (y_u^k - \mu_i)(y_u^k - \mu_i)^T}{\sum_k \zeta_k^t \sum_u \xi_{iu}^k} & \\
\mu_{ij} = \dfrac{\sum_k \zeta_k^t \sum_{uv} \xi_{iu,jv}^k y_{uv}^k}{\sum_k \zeta_k^t \sum_{uv} \xi_{iu,jv}^k}, & \Sigma_{ij} = \dfrac{\sum_k \zeta_k^t \sum_{uv} \xi_{iu,jv}^k (y_{uv}^k - \mu_{ij})(y_{uv}^k - \mu_{ij})^T}{\sum_k \zeta_k^t \sum_{uv} \xi_{iu,jv}^k} & \\
\mu_{B_1}^+ = \dfrac{\sum_k \zeta_k^t \sum_u \bar{\xi}_{iu}^k y_u^k}{\sum_k \zeta_k^t \sum_u \bar{\xi}_{iu}^k}, & \Sigma_{B_1}^+ = \dfrac{\sum_k \zeta_k^t \sum_u \bar{\xi}_{iu}^k (y_u^k - \mu_i)(y_u^k - \mu_i)^T}{\sum_k \zeta_k^t \sum_u \bar{\xi}_{iu}^k} & \\
\mu_{B_2}^+ = \dfrac{\sum_k \zeta_k^t \sum_{uv} \bar{\xi}_{iu,jv}^k y_{uv}^k}{\sum_k \zeta_k^t \sum_{uv} \bar{\xi}_{iu,jv}^k}, & \Sigma_{B_2}^+ = \dfrac{\sum_k \zeta_k^t \sum_{uv} \bar{\xi}_{iu,jv}^k (y_{uv}^k - \mu_{ij})(y_{uv}^k - \mu_{ij})^T}{\sum_k \zeta_k^t \sum_{uv} \bar{\xi}_{iu,jv}^k} &
\end{array} \tag{25}$$

The above equations are supposed to automatically discover the views of the training object instances through Eq.(23). However, our experiments show that directly using the E-M updates often results in inaccurate parameters of RARG components. This is because in the initial stage of learning, the parameters of each RARG component is often inaccurate, leading to inaccurate assignment probabilities (i.e. inaccurate view assignment). To overcome this problem, we can use a semi-supervised approach. First, the parameters of the RARG components are initially learned using view annotation data (view labels associated to the training images by annotators). Mathematically, this can be realized by fixing the assignment probabilities using the view labels during the E-M updates. For instance, if an instance k is annotated as view t, then we let $\zeta_k^t = 1$. After the initial learning process converges, we use the view update equation in Eq.(23) to continue the E-M iterations to refine the initially learned parameters. Such a two-stage procedure ensures that the parameters of a RARG component can be learned from the object instances with the view corresponding to the correct RARG component in the beginning of the learning process.

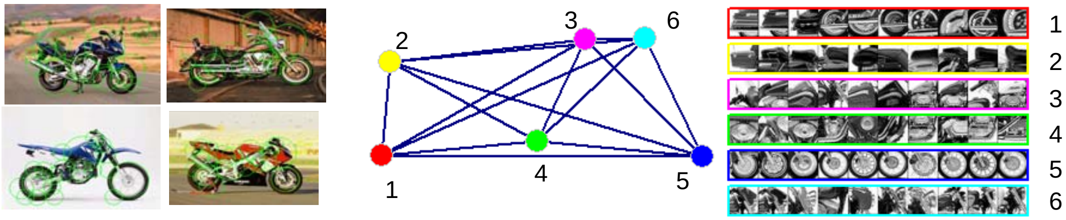


Figure 4: The RARG learned from the ’motorbike’ images.

4 Experiments

We compare the performance of our system with the system using the constellation model presented in [5]. We use the same data set, which consists of four object classes - *motorbikes*, *cars*, *faces*, and *airplanes*, and a common *background* class. Each of the classes and the *background* class is randomly partitioned into training and testing sets of equal sizes. All images are resized to have a width of 256 pixels and converted to gray-scale images. Image patches are detected by Kadir’s Salient Region Detector[7] with the same parameter across all four classes. Twenty patches with top saliency values are extracted for each image. Each extracted patch is normalized to the same size of 25×25 pixels and converted to a 15-dimensional PCA coefficient vectors, where PCA parameters are trained from the image patches in the positive data set. Overall, the feature vector at each node of the ARG is of 18 dimensions: two for spatial coordinates, fifteen for PCA coefficients, and one for the scale feature, which is an output from the part detector to indicate the scale of the extracted part. Feature vectors at the edges are the coordinate differences.

To keep the model size consistent with that in [5], we set the number of nodes in RARG to be six, which gives a good balance between detection accuracy and efficiency. The maximum size of the Association Graph therefore is 120 (6x20). But for efficiency, the Association Graph is pruned to 40 nodes based on the pruning criteria described in Section 2.1. In the learning process, we tried both inference schemes, i.e. GS+BO and LBP. But, in detection we only use GS+BO scheme because it is found to be more accurate. In LBP learning, relational features are not used because it is empirically found to result in lower performance. In GS+BO scheme, the sampling number is set to be proportional to the state space dimension, namely $\alpha \cdot 2 \cdot 40$ (α is set to 40 empirically). The *presence probability* r_i is computed only after the learning process converges because during the initial stages of learning, r_i is so small that it affects the convergence speed and final model accuracy. Besides, we also explore different ways of applying the background models in detection. We found a slight performance improvement by replacing B^+ with B^- in the detection step(Eq. (12)). Such an approach is adopted in our final implementation. Figure 4 shows the learned part-based model for object class ’motorbike’ and the image patches matched to each node. Table 1(next page) lists the object detection accuracy, measured by *equal error rate* (definition is in[5]), and the learning efficiency.

The most significant performance impact by our method is the improvement in learning speed - two times faster than the well-known method [5] if we use GS+BO for learning; five times speedup if we use LBP learning. Even with the large speedup, our method still achieves very high accuracy, close to those reported in [5]. The slightly lower performance for the *face* class may be because we extracted image patches in the lower resolution images. We found that small parts such as eyes cannot be precisely located in the images with a width of 256 pixels only. We decided to detect patches from low-resolution images because the patch detection technique from [7] is slow (about one minute for one original face image). The improved

Dataset	GS+BO	LBP	Oxford
Motorbikes	91.2%	88.9%	92.5%
Faces	94.7%	92.4%	96.4%
Airplanes	90.5%	90.1%	90.2%
Cars(rear)	92.6%*	93.4%	90.3%

Dataset	GS+BO	LBP	Oxford
Motorbikes	23i/18h	28i/6h	24-36h
Faces	16i/8h	20i/4h	
Airplanes	16i/16 h	18i/8h	40-100i
Cars(rear)	16i/14h	20i/8h	

Table 1: Object detection performance and learning time of different methods (xi/yh means x iterations and y hours). * Background images are road images the same as [5].

learning speed allows us to rapidly learn object models in new domains or develop more complex models for challenging cases.

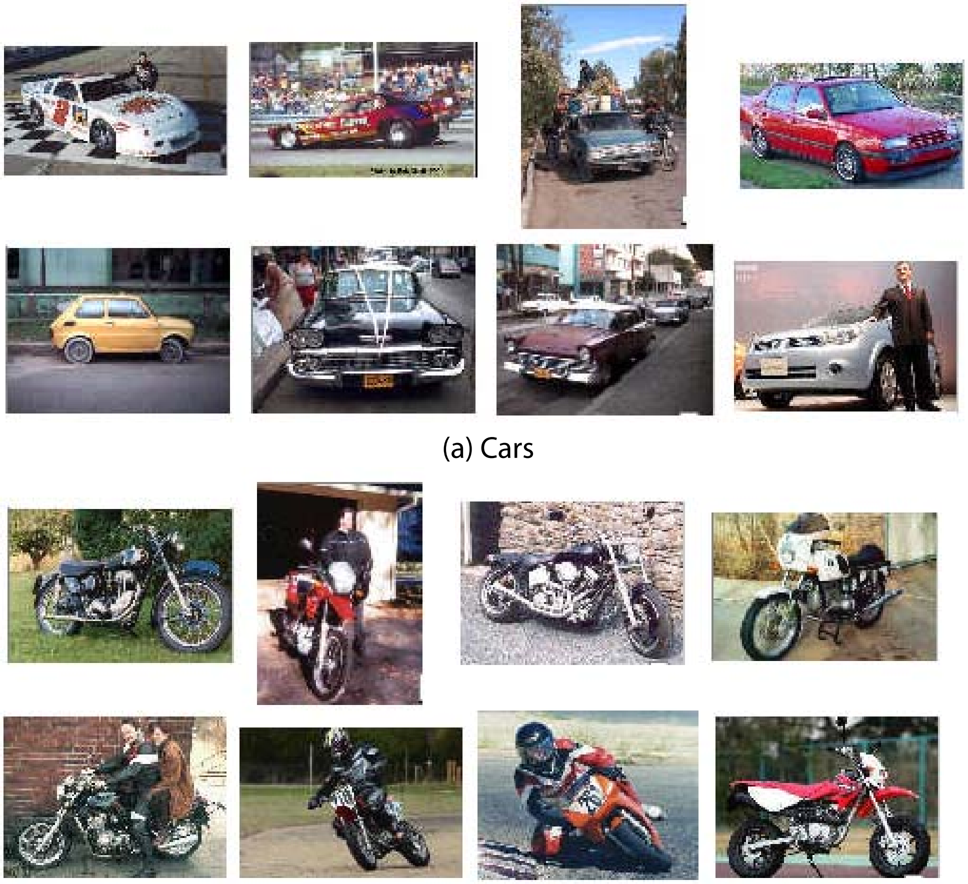
For multi-view object detection, we have built up our own data sets using google and altavista search engines. The data sets contain two object classes: 'cars' and 'motorbikes'. Each data set consists of 420 images. The objects in the data sets have continuous view changes, different styles and background clutters. The variations of the objects in the images roughly reflects the variations of the objects in web images, so that we can assess the performance of our algorithm for classifying and searching the web images. Before learning and detection, the images first undergo the same preprocessing procedures as the case of the single view detection. To save the computation cost, we only use two mixture components in the Mixture of RARG model. The performances using different learning schemes are listed below.

dataset	sing	manu	auto	relax
Cars	74.5%	73.5%	76.2%	76.3%
MotorBikes	80.3%	81.8%	82.4%	83.7%

Table 2: Multi-view object detection performance

The baseline approach is the single RARG detection ('sing'), namely we use one RARG to cover all variations including view changes. Three different multi-view learning methods are tested against the baseline approach. In learning based on automatic view discovery ('auto'), Eq.(23) is used to update the assignment probability (i.e. the view probability) in each E-M iteration. In learning based on manual view assignment('manu'), update through Eq.(23) is not used in E-M. In stead, the assignment probability is computed from the view annotation data and fixed throughout the learning procedure. In learning by combining view annotation and view discovery ('relax'), we first learn each component RARG model using view annotations. The automatic view discovery is then followed to refine the parameters. The view annotation procedure is realized by letting annotators inspect the images and assign a view label to each image. Here, because we only have two components, each image is assigned with either 'side view' or 'front view'. Although there are objects with 'rear view', they are very rare. Besides, we do not distinguish the orientations of the objects in 'side view'.

From the experiments, it is observed that the 'manu' mode performs worse than the 'auto' and 'relax' mode. This is because the continuous view variations in the data set makes the view annotations inaccurate. Overall, the 'relax' model performs best. This is consistent with our theoretical analysis: learning based on view annotation ensures the component RARGs can be learned correctly, and the following refinement by automatic view discovery optimizes the parameters of the component RARGs as well as view assignments which could be inaccurate by manual annotations.



(a) Cars

(b) Motorbikes

Figure 5: The multi-view objects for learning and testing

5 Conclusions

We have presented a new statistical part-based model, called RARG, for object representation and a new approach for object detection. We solve the part matching problem through the formulation of an Association Graph that characterizes the correspondences between parts in an image and nodes in the object model. We prove an important mathematical property relating the likelihood ratio for object detection and the partition functions for the MRFs defined over the Association Graph. Such discovery allows us to apply efficient variational methods such as Gibbs Sampling and Loopy Belief Propagation to achieve significant performance improvement in terms of learning speed and detection accuracy. We further extend the single RARG model to a mixture model for multi-view object detection, which improve the detection accuracy achieved by the single RARG model.

6 Acknowledgement

We thank Lexing Xie for inspiring discussions during our research, and helpful suggestions for part detection and model initialization.

References

[1] C. Andrieu, N. de Freitas, A. Doucet, and M. Jordan. An introduction to mcmc for machine learning. In *Machine Learning, vol. 50, pp. 5–43, Jan. - Feb.*, 2003.

[2] S. Ebadollahi, S.-F. Chang, and H. Wu. Automatic view recognition in echocardiogram videos using parts-based representation. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, pages 2–9, June, 2004.

[3] P. Erdos and J. Spencer. *Probabilistic Methods in Combinatorics*. Academic Press, 1974.

[4] P. Felzenszwalb and D. Huttenlocher. Efficient matching of pictorial structures. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, pages 66–75, 2003.

[5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, pages 66–73. IEEE, 2003.

[6] H.G.Barrow and R.J.Popplestone. Relational descriptions in picture processing. In *Machine Intelligence*, pages 6:377–396, 1971.

[7] T. Kadir and M. Brady. Scale, saliency and image description. *In International Journal of Computer Vision*, pages 45(2):83–105, 2001.

[8] S. Z. Li. A markov random field model for object matching under contextual constraints. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, pages 866–869, Seattle, Washington, June 1994.

[9] M. J. Wainwright and M. I. Jordan. Semidefinite methods for approximate inference on graphs with cycles. In *UC Berkeley CS Division technical report UCB/CSD-03-1226*, 2003.

[10] M. Weber, M. Welling, and P. Perona. Towards automatic discovery of object categories. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, pages 101–109. IEEE, 2000.

[11] M. Welling and Y. W. Teh. Belief optimization for binary networks: A stable alternative to loopy belief propagation. In *Uncertainty of Artificial Intelligence*, 2001, Seattle, Washington.

[12] J. S. Yedidia and W. T. Freeman. Understanding belief propagation and its generalizations. In *Exploring Artificial Intelligence in the New Millennium*, pages pp. 239–236,Chap. 8,, Jan. 2003.