Tensorize, Factorize and Regularize: Robust Visual Relationship Learning

Seong Jae Hwang[1] Sathya N. Ravi[1] Zirui Tao[1]
Hyunwoo J. Kim[1] Maxwell D. Collins[1] Vikas Singh[2,1]

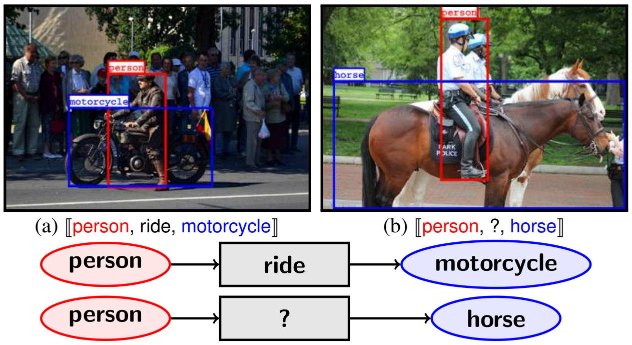
[1]Dept. of Computer Sciences, University of Wisconsin - Madison, Madison, WI
[2]Dept. of Biostatistics and Med. Informatics, University of Wisconsin - Madison, Madison, WI
http://pages.cs.wisc.edu/~sjh

Abstract

Visual relationships provide higher-level information of objects and their relations in an image – this enables a semantic understanding of the scene and helps downstream applications. Given a set of localized objects in some training data, visual relationship detection seeks to detect the most likely "relationship" between objects in a given image. While the specific objects may be well represented in training data, their relationships may still be infrequent. The empirical distribution obtained from seeing these relationships in a dataset does not model the underlying distribution well — a serious issue for most learning methods. In this work, we start from a simple multi-relational learning model, which in principle, offers a rich formalization for deriving a strong prior for learning visual relationships. While the inference problem for deriving the regularizer is challenging, our main technical contribution is to show how adapting recent results in numerical linear algebra lead to efficient algorithms for a factorization scheme that yields highly informative priors. The factorization provides sample size bounds for inference (under mild conditions) for the underlying ⟦object, predicate, object⟧ relationship learning task on its own and surprisingly outperforms (in some cases) existing methods even without utilizing visual features. Then, when integrated with an end-to-end architecture for visual relationship detection leveraging image data, we substantially improve the state-of-the-art.

1. Introduction

The core primitives of an image are the objects and entities that are captured in it. As a result, a strong thrust of research, formalized within detection and segmentation problems, deals with accurate identification of such entities, given an image. On the other hand, there is consensus that to "understand" an image from a human's perspective [29, 21], higher-level cues such as the relationship between the objects are critical. Being able to reason about which entities are likely to *co-occur* [32, 25] and *how they interact* [54, 12] is a powerful mid-level feature that endows a system with auxiliary information far beyond what individual object detectors provide. Starting with early work on AND-OR graphs [31, 27] and logic networks [46, 43], algorithms which make use of relational learning are becoming mainstream within vision, offering strong performance on categorization and retrieval tasks [1, 13]. Furthermore, many interesting applications [9, 51, 4] have begun to appear as richer datasets become available [4, 29, 55, 52].



(a) ⟦person, ride, motorcycle⟧ (b) ⟦person, ?, horse⟧

(c) Objects interactions with predicates

Figure 1: **Visual relationship detection:** the goal is to infer visual relationships that best describe the interactions among those objects. (a): A relationship instance in a training set. (b): An unknown relationship to predict. (c): The interactions of the objects (i.e., motorcycle and horse are both 'ridable') can be used to infer the correct relationship.

An intuitive visual relationship learning setup is as follows. Given a sufficiently large set of images where the objects have been localized, we process the images and specify the "relationship" between the objects; for example, person and couch related by sitting on and/or person and bike related by riding. Then, with a learning module in hand, it should be possible to *learn* these associations to facilitate concurrent estimation of the object class *as well as* their relationship. For instance, a model may suggest that given a high confidence for the bike class, a smaller set of classes for the other object are likely, and perhaps, a small set of relationships may explain the semantic association between those two objects. The authors in [40] showed that this idea of "Visual Phrases" performs well even when provided with

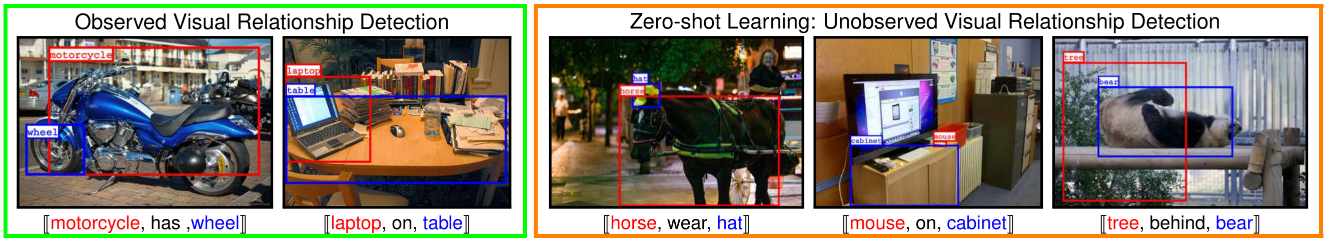


Figure 2: Examples of visual relationships detected by our algorithm given objects and their object bounding boxes. The left two relationships (**green box**) were observed in the training set. The right three relationships (**orange box**) *not* observed in the training set are potentially much harder to detect.

a small set of 13 common relationships. However, for such a learning task to work, the training data size should be sufficient to cover all possible relationships. But as we seek for a richer dataset, the challenge comes from its availability and the skewed relationship distribution.

Last year, [23] presented a visual relationship dataset, Visual Genome, to help research on this topic: over 100K images with 42K unique relationships. Visual Genome is a massive expansion of the Scene Graph dataset [21] (gives an image as a first-order network of its objects (vertices) and their visual relationships (edges)). Visual Genome connects the individual scene graphs to one another based on their common objects and/or relationships encoding the interconnectedness of many complex object interactions.

From Visual Phrases to Scene Graph Prediction. Given a set of detected objects (i.e., person, dog, phone objects) in an image and possible predicates (i.e., on, next to, hold predicates), the goal is to infer the most likely relationships (i.e., ⟦person, hold, phone⟧ relationship) among the objects, see Fig. 2. The Visual Phrases based algorithm [40] builds a model for *each* unique relationship instance to fully detect all possible relationships, i.e., # of predicates × # of object categories[2]. Independent object-wise predictions are combined using a decoding scheme that takes all responses and then decides on the final image-specific outcome. The formulation is effective but as noted by [29], it becomes infeasible as the number of unique relationships (⟦object, predicate, object⟧) exceed several thousands – as is the case in the new Visual Genome dataset. To address this limitation, in [29] the authors propose building "joint" models that do *not* enumerate the set of all relationships and instead are proportional to the number of object categories plus predicates. This set is much smaller and effective to the extent that these fewer degrees of freedom capture the large number of relationships. As discussed in [29], often the language prior can compensate for such disparity between the model complexity and dataset complexity but also suffers if the semantic word embeddings fall short [5]. Recently, as a natural extension to the individual relationship detection, understanding an image at a broader scope as a *scene graph* [52] has been proposed where the goal is to infer the entire interconnectedness of the objects (nodes) in the image with various visual relationships (edges). While the detection on objects and relationships 'help' each other, relatively more challenging visual relationship inference is often the bottleneck within such combined approaches.

Key Components for Improved Visual Relationship Learning. A hypothetical model may offer improvements in visual relationship learning if it has the following properties: **(1)** Leaving aside empirical issues, the model complexity (i.e., degrees of freedom) should be able to compensate for the complexity of the data (i.e., number of object categories) while still guaranteeing performance gains for the core learning problem under mild assumptions. **(2)** Additionally, it would be desirable if the above characteristic can also generalize to unseen data (i.e., relationships *not* in training data) with little information about unseen observations (i.e., *unknown* category distributions).

Contributions. (i) We view visual relationship learning as a slightly adapted instantiation of a multi-relational learning model. Despite its non-convex form, we show how recent results in linear algebra yield an efficient optimization scheme, with some guarantees towards a solution. **(ii)** We derive sample complexity bounds which demonstrate that despite the ill-posed nature, under sensible conditions, inference can indeed be performed. This scheme yields powerful visual relationship priors despite the extremely sparse nature of the data. **(iii)** Our proposal integrates the priors with an adaptation of visual relationship detection architecture. This end-to-end construction brings the best performance of the much more challenging scene graph prediction tasks [52] on the Visual Genome dataset by modulating the deep neural network structure with a provably stable relational learning module. The key leverage comes from overcoming the sparsely observed visual relationships (∼2% of possible relationships) with contribution (i)-(ii).

2. Relational Learning in Vision

In this section, we briefly review some of the related works. In the past years, low-to-mid level computer vision tasks have seen a renaissance leading to effective algorithms [24, 35] and various datasets [28, 4, 55]. Building upon these successes, higher-level tasks, such as scene understanding [14, 56] and relationship inference [26, 29, 50, 52], which often rely on the lower-level modules are being more intensively studied. In particular, inferring the *visual relationship* between objects is the next logical goal – going from object level detection to semantic relations among objects for higher-level relationships. For instance, sim-

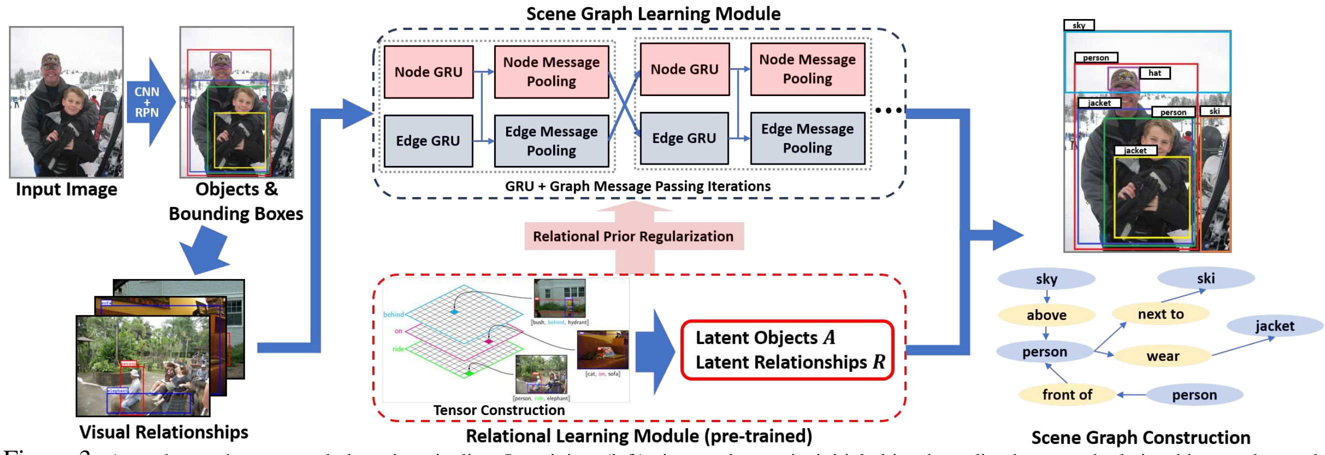


Figure 3: An end-to-end scene graph detection pipeline. In training, (left) given an image, its initial object bounding boxes and relationships are detected. Then, (top middle) its objects and relationships are estimated via scene graph learning module [52]. Our pre-trained (thus not being trained in this pipeline) tensor-based relational module (bottom middle) provides a visual relationship prediction as a dense relational prior to refine the relationship estimation which also regulates the learning process of the scene graph module. In testing, (right) the scene graph of an image is constructed based on both modules.

ple contextual features such as co-occurrence [25, 32] are useful but not rich enough for detailed semantic relationship among objects such as those required within VQA [4]. On the other hand, human-specific relationships based on human-object interaction [39, 54], while expressive, limit the scope of information inferable from natural images containing many types of objects. From a different perspective, inferring visual information from images under various assumptions (i.e., in the wild) has been utilized to retrieve task-specific visual information as well [36, 45].

A deeper understanding of images is being successfully demonstrated in various semantic inference tasks. For instance, answering abstract questions related to a given image called visual question answering (VQA) [4] has shown good results [55, 3] with the availability of various datasets [4, 55]. Also, image captioning [10, 53] can infer detailed high-level knowledge from image.

In this paper, we focus on inferring a mid/high level description commonly referred to as *visual phrases* [40, 23] that provides systematically structured visual relationships (i.e., person rides a car as ⟦person, ride, car⟧) that is both quantifiable and expressive. For instance, understanding an image in terms of the objects *and* their visual relationships has been recently formulated as a *scene graph detection* [52] based on the large-scale Visual Genome dataset [23] which requires simultaneously performing both higher-level visual phrase inference and lower-level object recognition. As seen on the right of Fig. 3, a successfully constructed scene graph provides rich context about the image for an upstream system-level model (i.e., VQA) where the bottleneck often comes down to semantic ambiguities and sparse sample observations.

3. Collective Learning on Multi-relational Data

Much of our technical development will focus on distilling the sparsely observed relationship data towards a precise regularization that will be integrated into an end-to-end pipeline. To setup our presentation for deriving this prior, we first briefly describe encoding/representing the data and then obtain an objective function to model the inference task for the Relational learning module in Fig. 3.

Tensor Construction. Suppose we are given a dataset of N images that contains n object categories and m possible predicates which are both indexed. For instance, an image can have an object $i \in \{1, \ldots, n\}$ having a predicate $k \in \{1, \ldots, m\}$ with another object $j \in \{1, \ldots, n\}$. We can construct a relationship tensor $X \in \mathbb{R}^{n \times n \times m}$ where $X(i, j, k)$ contains the number of occurrences of the i'th object and j'th object having k'th predicate in the dataset. If the relationship of person (object index i) and bike (object index j) described by ride (predicate index k) has shown up p times, then we assign $X(i, j, k) = p$. We can also think of X as a stack of m matrices $X_k \in \mathbb{R}^{n \times n}$ for $k \in \{1, \ldots, m\}$: each X_k contains information about the k'th predicate among all the objects in the data (see Fig. (4)). Note that in practice, only a small fraction (i.e., $\sim$1%) of the possible relationships are observed out of mn^2 possible relationships; the relationship tensor is extremely sparse.

Why Tensor Construction? In multi-relational learning such as visual relationship learning, it is critical to appropriately represent the inter-connectedness of the objects [42, 16]. Such multi-relational information of any order can be easily encoded as a higher order tensor where its construction does not require any priors (parametric distributions in Bayesian Networks [15]) or assumptions (Markov Logic network structure [38]). Our main motivation is: even though the objects are represented as points in $\mathbb{R}^n$, due to the sparse matrix slices X_k's, we may assume that the objects are embedded in fewer dimensions $r<n$. In principle, this can be accomplished by a message passing module [52] within the pipeline shown in Fig. 3 but experimentally, we find that concurrent learning both modules is challenging.

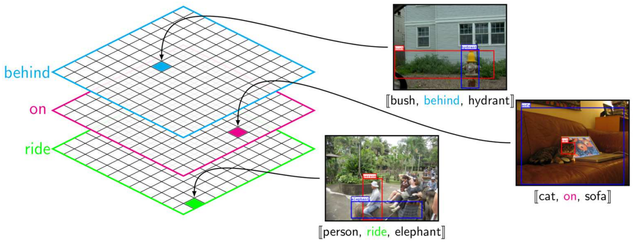


Figure 4: Muti-relational tensor $X \in \mathbb{R}^{n \times n \times m}$ given n object categories and m possible predicates. The value at $X(i, j, k)$ is the number of ⟦i'th object, k'th predicate, j'th object⟧ instances observed in the training set. Due to the sparse nature of the relationship instances, only $\sim 1\%$ of the tensor constructed from our training set has non-zero entries.

Why not other Tensor Decompositions? Recently, many authors [2, 19] have shown that learning latent representations correspond to decomposing a tensor into low-rank components. While many standard techniques [18, 48] exist, they are inappropriate for multi-relational learning for a few reasons. For instance, polyadic decomposition [18] puts rigid constraints on the relational factor (i.e., diagonal core tensor) which is counterintuitive in relational learning [33]. Ideally, we want the *converse* construction where the relational factors are flexible with respect to the "latent" object representations. In that sense, our model is similar to the less widely used Tucker 2−decomposition [48], but Tucker 2 allows too many degrees of freedom on the objective factor. Second, the solution of typical solvers [18, 48] is often not unique. This is not relevant in many factor analysis tasks that do *not* rely on the representations (i.e., Eigenfaces [49]), but this property is undesirable in our formulation where we explicitly consider the relationships among the objects in their "latent" representations. In other words, two equally optimal solutions could interpret the same relationship differently. Thus, we need to impose consistency in representations by identifying a unique solution (via additional regularization).

In this section, we describe a novel relational learning algorithm which addresses the above issues and provides the generalization power needed for visual relationship detection. We first explain our model motivated by a three-way collective learning model [33] which derives a set of latent object representations connected by relational factors. Later, we extend this formulation and describe our relationship inference model which guarantees a unique solution for consistent objects representations and their relationships. We then empirically show how our pipeline (Fig. 3) integrating the regularization (or prior) obtains benefits.

3.1. Three-way Relational Learning

Recall our mild assumption that the objects can be represented in a lower dimensional space with dimensions $r < n$. We will now explain our model in two steps: first, given the multi-relationship tensor X, our goal is to derive the latent representation of its objects $A \in \mathbb{R}^{n \times r}$ of rank r; secondly, assuming that we know the lower dimensional representation A of the objects, now we can define the relationship-specific factor matrix $R_k \in \mathbb{R}^{r \times r}$ for each $k \in \{1, \ldots, m\}$ for each relationship matrix X_k. Observe that A is common across all the relationships where the i'th row of A is the latent representation of the i'th object as desired. On the other hand, each factor matrix R_k individually corresponds to the k'th relationship and constitutes its respective matrix X_k (see Fig. (4)) with the common latent representation A. We can now write our model as,

$$X_k \approx A R_k A^T. \tag{1}$$

Hence, our optimization problem to solve is,

$$\min_{A, R_k} \sum_{k=1}^{m} ||X_k - A R_k A^T||_F^2 \tag{2}$$

where we will learn A and R_k's simultaneously. Such a decomposition of a three dimensional tensor is referred to as Tucker 2−decomposition [22]. The "2" refers to the fact that we are learning two "types" of matrices in some sense.

Now we discuss a crucial property of the tensor X that is very relevant. Observe that since a relationship and its converse (i.e., person on bike and bike on person) need *not* always occur together, each X_k is not always symmetric, thus preventing us from effectively using many readymade tools from matrix analysis like the spectral theorem, eigendecomposition and so on. In our multi-relational tensor X, a predicate often cannot be sensibly applied in the other direction. Thus, we propose alternative strategies that includes certain reformulations. Before we present our final algorithm to solve problem (2), we will show how certain reformulations will enable us to design efficient algorithms.

A possible solution strategy to solve the above formulation (2) is using a conventional approach such as the Alternating Least Squares (ALS) method. In this method, one variable is optimized while fixing all the other variables. *Importantly, for the ALS algorithm to be efficient, we need all of the optimization subproblems to be easily solvable.* However, note that solving for A while fixing R_k's is not easy since it involves fourth order polynomial optimization.

4. Algorithm

In this section, we present our algorithm (Alg. 1) consisting of a novel initialization scheme followed by an iterative scheme to solve our the multi-relational problem (2) with an additional regularization term that is weakly derived from [47]. Then, we show how the algorithm can be integrated into the formulation in Fig. 3 as the Relational learning (RL) module which provides a dense predicate prior.

4.1. Multi-relational Tensor Factorization

To make our analysis easier, as the first step, we use auxiliary variables to decouple A and A^T in the objective function resulting in a method of multipliers type formulation,

$$\min_{A, R_k} \sum_{k=1}^{m} ||X_k - B_k A^T||_F^2 \quad \text{s.t.} \quad B_k = A R_k. \tag{3}$$

Algorithm 1 Alternating Block Coordinate Descent on (8)

```
Given: $X \in \mathbb{R}^{n\times n\times m}$, $X_k := X(:,:,k)$, rank $r > 0$
Low-rank Initialization:
$\overline{X} \leftarrow \sum_{k=1}^{m} X_k$
$\overline{U}\overline{\Sigma}\overline{V}^T \leftarrow$ SVD$(\overline{X}, r)$
$A \leftarrow \overline{V}\overline{\Sigma}^{1/2}$
for $k = 1, \ldots, m$ do
    $B_k \leftarrow \overline{U}\overline{\Sigma}^{1/2}$
    $R_k \leftarrow (A^TA)^{-1}(A^TX_kA)(A^TA)^{-1}$
end for
Iterative descent method:
while Convergence criteria not met do
    $A \leftarrow$ gradient descent on (8) w.r.t. $A$
    for $k = 1, \ldots, m$ do
        $B_k \leftarrow$ gradient descent on (8) w.r.t. $B$
        $R_k \leftarrow \left(A^TA\right)^{-1}\left(A^TB_k\right)$
    end for
end while
Output: $A \in \mathbb{R}^{n\times r}$, $B_k \in \mathbb{R}^{n\times r}$, $R_k \in \mathbb{R}^{r\times r}$ for $\forall k$
```

For the purpose of designing an algorithm, let us analyze only the objective function in (3) ignoring the equality constraints resembling matrix factorization by letting $m = 1$:

$$\min_{A,B} ||X - BA^T||_F^2. \tag{4}$$

It is easy to see that the above problem can be solved exactly using the Singular Value Decomposition (SVD) of X. When $m > 1$, we need to identify matrix factorization type models where SVD (or something related) serves as a subroutine. Recent works use SVD as a subroutine in primarily a few different ways to solve problems that can be posed as matrix factorization problems: preprocessing step [8] at each iteration [20] and thresholding schemes [6]. Intuitively, in the above works, the SVD of an *appropriate* matrix (chosen specifically depending on the problem context) provides a good estimate of the global optimal solution of rank constrained optimization problems both theoretically [41] and practically in vision applications [30]. Essentially, these works show that with a specially constructed matrix, having an initialization already gets close to optimal solutions, and then any descent method is guaranteed to work. Unfortunately, these results do *not* extend to our case when $m > 1$. we generalize this idea, derive sample complexity bounds on the number of predicates needed to learn the latent representations and give an efficient algorithm.

Low-rank Initialization via SVD. For a given generic $X \in \mathbb{R}^{n\times n}$, (4) can be solved by

$$A = V\Sigma^{1/2}, \quad B = U\Sigma^{1/2} \tag{5}$$

where $U\Sigma V^T = X$ is the SVD of X. Under certain conditions, recent works such as [44] and [47] have shown that an initial point for other common low-rank decomposition formulations can be estimated within the "basin of attraction" to guarantee the globally optimal solution; hence this provides the exact latent representation of objects.

Since our formulation consists of multiple X_k and B_k for $k \in \{1, \ldots, m\}$, we must construct an appropriate matrix that will *provably put us in the basin of attraction*. Intuitively, let us assume that the matrices X_k are sampled from an underlying distribution with the mean $\mathbf{E}(X)$. In order to get an unbiased estimator of the mean, we merge all X_k for $k \in \{1, \ldots, m\}$ into $\overline{X} = \sum_{k=1}^{m} X_k$. So, as a heuristic, we may initialize the low-rank latent representation A by setting $A = \overline{V}\overline{\Sigma}^{1/2}$ where $\overline{U}\overline{\Sigma}\overline{V}^T = \overline{X}$ represents the SVD of $\overline{X}$. So, is this seemingly ad-hoc heuristic averaging justified? We now show when our initialization is guaranteed to be good under the mild assumption that each predicate is *independent* and is drawn from an underlying distribution.

Lemma 4.1.1. *Let $\mathbf{E}(X)$ be the true abstract object relationship matrix from which X_k's are sampled from, $\epsilon > 0$ be the error of our estimate and $\delta > 0$ be the failure probability. Furthermore, assume that each X_k for $k \in \{1, \ldots, m\}$ is an independent bernoulli random matrix. Then A is an (ϵ, δ) solution if $m = \mathcal{O}\left(\frac{1}{\epsilon}\log\left(\frac{n}{\delta}\right)\right)$. (Proof in supplement)*

Remark. For a fixed $\epsilon > 0$ and $\delta > 0$, the number of predicates required for an accurate estimation of the latent representation A has only a logarithmic dependence on the number of objects n suggesting that our procedure needs a small number of predicates to find the latent representation of objects as we also see in practice. Integrating the prior derived by this formulation within the pipeline in Fig. 3 to do the full inference concurrently offers no simple way for the optimization scheme to exploit the nice algebraic and statistical properties we use here.

Having initialized A, we simply set $B_k = \overline{X}A(A^TA)^{-1}$ as the initial point. Another option is to use the least squares solution $B_k = X_kA(A^TA)^{-1}$ with respect to each X_k, but this has a higher chance to overfit the data. Finally, each $R_k \in \mathbb{R}^{r\times r}$ for $k \in \{1, \ldots, m\}$ can be solved with its respective X_k given the original factorization setup (2):

$$R_k = (A^TA)^{-1}(A^TX_kA)(A^TA)^{-1}. \tag{6}$$

Alternating Block Coordinate Descent. Similar to Section 4.1, let us first consider problem (4). We see that (4) has multiple global optimal solutions since the value of the loss is invariant to a basis transformation: $B' = BP$ and $A' = AP^{-T}$ for any invertible matrix $P \in \mathbb{R}^{r\times r}$ has the same objective function value as B and A. Thus, we add a term that restricts such degenerate cases:

$$\lambda_p \sum_{k=1}^{m} ||A^TA - B_k^TB_k||_F^2 \tag{7}$$

where $\lambda_p > 0$. A high value of λ_p, makes the two factors B_k and A to be on the unit "scale", or in other words, the factors are normalized [47]. Our final model which adds the regularization in (7) to a formulation equivalent to (3) is

$$\min_{A,R_k,B_k} \sum_{k=1}^{m} ||X_k - B_k A^T||_F^2 + \gamma \sum_{k=1}^{m} ||B_k - AR_k||_F^2 + \lambda_p \sum_{k=1}^{m} ||A^T A - B_k^T B_k||_F^2. \tag{8}$$

Equivalence means that there exists some $\gamma > 0$ such that the optimal solutions of (3) and (8) coincide, a direct consequence of Lagrange multiplier theory [7]. Note that the dual variable γ controls the fit to the constraint $B_k = AR_k$, so we will apply a continuation technique to solve (8) (without (7) for now) for increasing γ to enforce $B_k = AR_k$ [34]. Then, we fix γ and add (7) to solve (8). We used $\lambda_p = 0.01$.

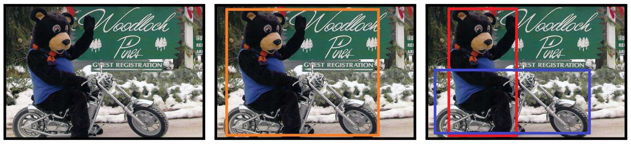
Solving for a Fixed γ. We iteratively solve for A and each B_k for $k \in \{1, \ldots, m\}$ individually with gradient descent methods as follows at each iteration. First, to solve A, we fix B_k for $k \in \{1, \ldots, m\}$ and perform gradient descent with respect to A as in line 12 of Alg. 1. Second, to solve each B_k, we fix A and $B_{\bar{k}}$ for $\bar{k} \neq k$ and perform gradient descent with respect to B_k as in line 14 of Alg. 1. To solve both of these subproblems, we used Minfunc/Schmidt solver with backtracking line search.

Note that we can solve each R_k for $k \in \{1, \ldots, m\}$ in a closed form $R_k = \left(A^T A\right)^{-1} \left(A^T B_k\right)$ since the last term does not involve any R_k (line 15 of Alg. 1). The optimization problem to solve for B_k and R_k is decomposable, so one main advantage is that they can be solved in parallel. The above procedure produces a monotonically decreasing sequence of iterates thus guaranteeing convergence [17]. In the supplement, we describe how the procedure can be thought of as a "meta" algorithm for factorization problems in vision which may be of independent interest.

4.2. Scene Graph Prediction Pipeline

We now describe the training procedure of the pipeline (Fig. 3). See supplement for other low-level details.

Relational Learning Module. We first setup the RL module by constructing the multi-relational tensor $X \in \mathbb{R}^{n \times n \times m}$ on the Visual Genome dataset as described before. Then, for $r = 15$, we solve for the latent representation of the objects $A \in \mathbb{R}^{n \times r}$ and the factor matrices $R_1, \ldots, R_m \in \mathbb{R}^{r \times r}$ based on (8) as in Alg. 1. Next, using the trained A and $R_1, \ldots, R_m$, we reconstruct the low-rank multi-relational matrix $\hat{X}$ which is the stack of m low-rank relational matrices similar to X except that each slice is $\hat{X}_k = AR_k A^T$ for $k \in \{1, \ldots, m\}$. Then, given objects i and j, the predicted predicate distribution is $k_{RL} = softmax(\hat{X}(i, j, :)) \in \mathbb{R}^m$.



(a) Predicate (b) Phrase (c) Relationship

Figure 5: **Detection Task Conditions:** Given object bounding boxes: (a) Predicate (easy): does not require bounding boxes. (b) Phrase (moderate): requires relationship bounding box (orange) containing both objects. (c) Relationship (hard): requires individual bounding boxes (red/blue).

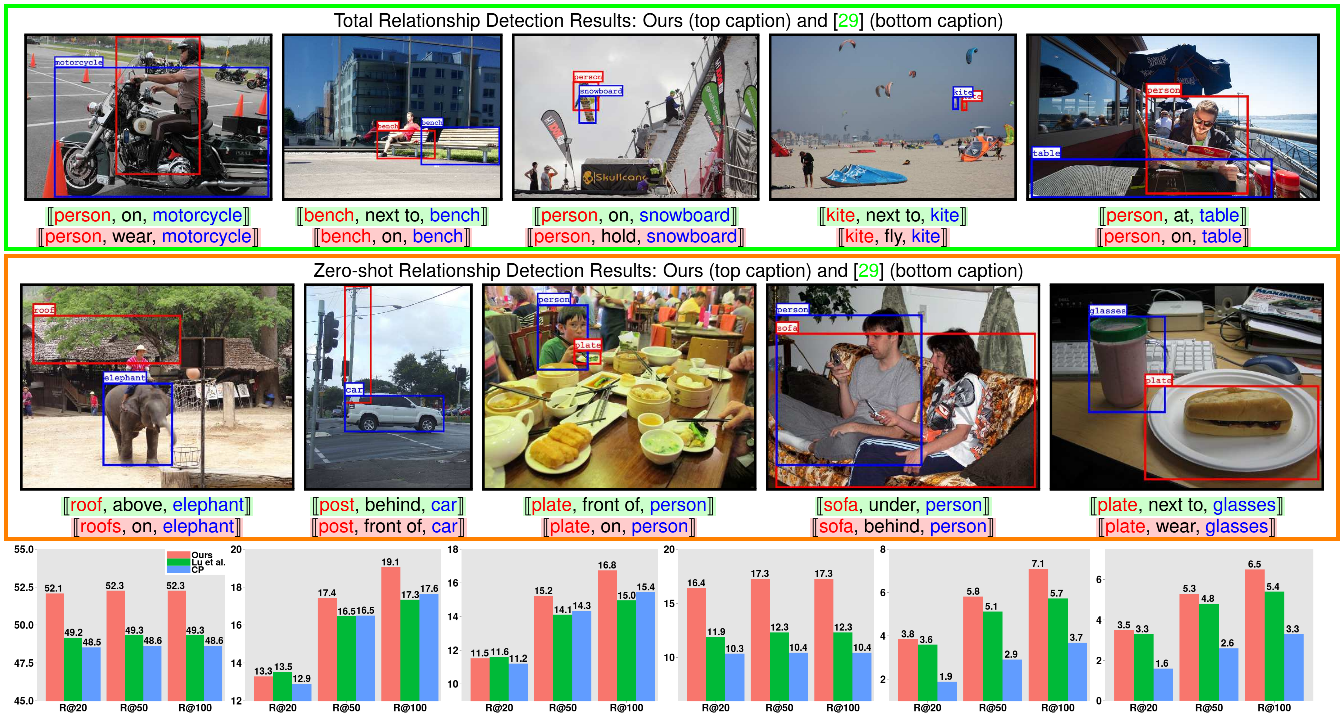
Training the Pipeline. Given an image, the initial object bounding boxes are detected via a Region Proposal Network [37] to train our end-to-end pipeline for scene graph prediction which consists of two modules (see Fig. 3). (a) **Scene graph (SG) module**: The iterative message passing network by [52] predicts both objects and predicates concurrently. (b) **Relational learning (RL) module**: Our tensor-based relational learning provides predicate prior $\hat{X}(i, j, :)$ between object i and j where the low-rank tensor is now constructed based on the entire Visual Genome training set. Given object-subject bounding boxes, our pipeline trains its relationship as follows: (1) The SG module estimates the object labels i^* and j^* along with the predicate distribution $k^*_{SG} \in \mathbb{R}^m$. (2) The RL module computes the predicate prior based on those estimates: $k^*_{RL} = softmax(\hat{X}(i^*, j^*, :)) \in \mathbb{R}^m$. (3) The prior k^*_{RL} is randomly applied to the network-based estimate k^*_{SG} as

$$k^* = k^*_{SG} \odot D(k^*_{RL}, \theta) \tag{9}$$

where $\odot$ is the Hadamard product and $D(y, \theta) \in \mathbb{R}^m$ is a 'y-or-1' filter where the i'th element is $y(i)$ with probability θ or 1 with probability $1 - \theta$. Intuitively, each entry of the SG module predicate prediction k^*_{SG} has a θ chance to be regularized by the RL prior k^*_{RL} via scaling. In practice, this either increases underestimated scores (i.e., rare object combinations) or decreases overestimated scores where the frequency of influence is determined by θ (i.e., higher θ regularizes more often). (4) Using the new predicate prediction k^*, relationship loss is now computed and backpropagated to the SG module with respect to both objects *and* predicates. During the training, in order to avoid the SG module prematurely converging because the pre-trained RL module does most of the heavy lifting, we first do a 'warm start' where we train the SG module *without* the RL module ($\theta = 0$) for 100K iterations. Then, we include the RL module for extra 50K iterations with $\theta = 0.2$ which was chosen empirically from $\theta \in \{0.2, 0.4, 0.6\}$ after observing that high θ values were less effective.

5. Experiments

We evaluate our model on two datasets. First, we test our regularization model described in Sec. 4.1 as a standalone method on the Scene Graph dataset [21] and compare against the relationship detection method by Lu et al. [29]. To show that performance gains are *not* just from the decomposition formulation (1), we also compare against Tucker 2 [48] and PARAFAC [18]. Second, for more difficult scene graph prediction tasks on Visual Genome [23], we show significant improvements over the recent state-of-



(a) Total Predicate (b) Total Phrase (c) Total Relationship (d) ZS Predicate (e) ZS Phrase (f) ZS Relationship

Figure 6: The total visual relationship detection (top row in green box) and the zero-shot visual relationship detection results (middle row in orange box) on Scene Graph dataset using our algorithm (top caption) and [29] (bottom caption). The correct and incorrect predictions are highlighted in green and red respectively. Visual relationship detection results (bottom row) on Scene Graph using ours (**red**), Lu et al. (**green**) and CP (**blue**). Best viewed in color.

the-art message passing network model by Xu et al. [52] using our end-to-end pipeline that integrates our tensor-based relational module with their message passing model [52]. The dense prior inferred from our provably robust relational module directly influences both the training and testing of the pipeline in a holistic manner as shown in Fig. 3. In both evaluations, we measure the true positive rate from the top p confident predictions referred to as recall at p (R@p) since not all ground truth labels can be annotated.

5.1. Scene Graph Dataset

We used the same set of 5000 training (<1% unique tuples) and 1000 test images with $n = 100$ object categories and $m = 70$ predicates as in [29].

Visual Relationship Prediction Setup. The procedure of constructing the low-rank multi-relational matrix $\hat{X}$ is identical to the description in Sec. 4.2 where in this case we use the Scene Graph dataset. Then, the predicted predicate between object i and j is $k^* = \text{argmax}_k\, \phi_{ij}\hat{X}(i,j,k)$ based on a vector of 'probability distribution' of predicates where ϕ_{ij} is a weight based on a simple word-vector distance between the categories i and j.

Prediction Tasks. We setup three different prediction experiments of varying difficulties (see Fig. 5 and supplement for details): **(a)** Predicate, **(b)** Phrase and **(c)** Relationship predictions. These are performed at R@p for $p \in \{100, 50, 20\}$ in two settings: **(1)** Total and **(2)** Zero-shot (test set *not* observed in training).

5.2. Visual Genome Dataset

We used the cleaned up version of the dataset following [52] to account for poor/ambiguous annotations which consists of $108,077$ images 25 objects and 22 relationships where we used 70% for training and 30% for testing. For the experiments, we used the most appearing $n = 150$ object categories and $m = 50$ predicates (11.5 objects and 6.2 relationships per image on average).

Prediction Setup. Once the pipeline is trained following Sec. 4.2, the prediction result is simply the forward propagation output of the pipeline except we now set $\theta = 1$ to fully use the relational prior k^*_{RL}.

Scene Graph Prediction Tasks. Detecting a scene graph requires inference on three parts: predicate, object class and bounding box which requires accurate predictions on these parts incrementally [52] as shown in Table 1. For all these tasks, we used R@p for $p \in \{100, 50, 20\}$.

5.3. Results on Relationship Learning Tasks

Visual Relationship Detection on Scene Graph. We show visual relationship detection results on the Scene Graph dataset using CP [18], Lu et al. [29] and our algorithm (Tucker [48] results in supplement) at the bottom of Fig. 6. For all tasks, our results outperform other methods. Especially, our zero-shot prediction (Fig. 6 (d)) results substantially outperform the state-of-the-art ([29]) by $\sim$ 40% in all recalls. In much more difficult phrase (b,e) and relationship (c,f) detection (Fig. 6), we achieve improvements in all tasks under almost all recalls. We observe that our *zero-shot*

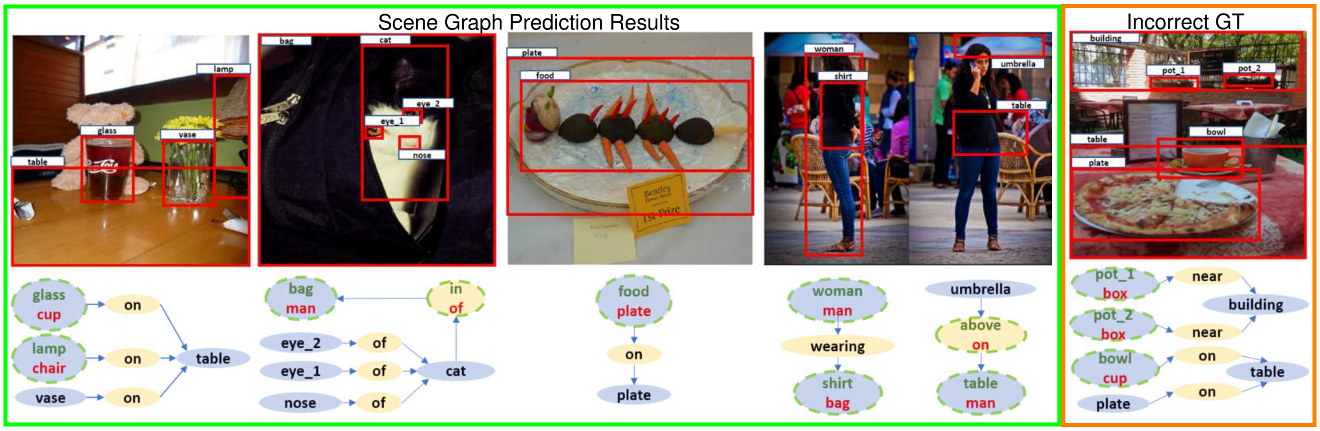
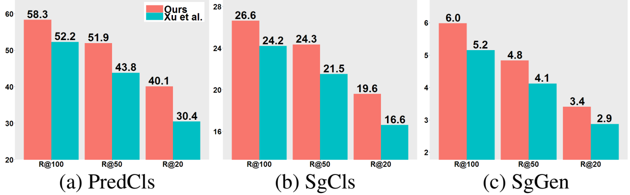


Figure 7: Scene graph classification results on Visual Genome using ours and [52]. For each column, the predicted objects (blue ellipses) and their relationships (yellow ellipses) are constructed as a scene graph its top image. The bounding boxes labels reflect our prediction results. For difficult predictions (**green dashed boundary**) where our model has correctly predicted (top **green**) and while [52] has misclassified (bottom **red**) are shown. The rightmost column is an example of a case where our model provides more accurate predictions (**pot** and **bowl**) than those of the ground truth (**box** and **cup**).

Prediction Tasks	Predicate	Object	B-box
Predict Predicate (**PredCls**)	✓		
Classify SG (**SgCls**)	✓	✓	
Generate SG (**SgGen**)	✓	✓	✓

Table 1: Scene graph detection tasks. Check marks indicate required prediction components. The tasks become incrementally more demanding from top (PredCls) to bottom (SgGen).



(a) PredCls (b) SgCls (c) SgGen

Figure 8: Scene graph detection task (see Table 1) results on Visual Genome using ours (**red**) and [52] (**cyan**). Our pipeline *without* the RL module show results similar to [52] (**cyan**).

predicate detection results (Fig. 6 (d)) given *known* object pairs is competitive with the *total* phrase detection results by [29] (Fig. 6 (b)) given *unknown* object pairs. This implies that while accurate object detection is crucial for visual relationship detection, more difficult zero-shot learning is a *less* critical factor for our algorithm.

Scene Graph Prediction on Visual Genome. We now show the scene graph prediction results (Fig. 8) on Visual Genome using Xu et al. [52] and our pipeline (Fig. 3). We also evaluated [29] on the same tasks, but the model did not scale well to the task complexity so the performances were lower than the other two methods by large margins (see supplement for full comparisons). **(a)** PredCls: Our model provides significant improvements in the predicate detection tasks in all recalls by at most ∼30% in R@20. Since this task only demands predicate predictions, such large improvements demonstrate that the tensor-based RL module functions as an effective prior for inferring visual relationships by better utilizing the large but sparse dataset. **(b)** SgCls: The results on the scene graph classification (Fig. 8(b)) show that our model improves object classifications as well in all recalls where our R@50 result is on par with R@100 of [52]. The boost in predicate prediction improves overall inference on the interconnected object and predicate inference of the SG module [52] during the training. **(c)** SgGen: On the last task which also predicts the bounding box, our model showed ∼10% improvements in all recalls over [52].

Remarks. We observe that our RL module provides boosts on not only the predicate detection (PredCls) but also the interdependent object classification tasks (SgCls and SgGen) enabled by our composite pipeline (Fig. 3), and this is our initial hypothesis: relationship learning is a bottleneck which needs to be focused on. Second, as seen in the rightmost column of Fig. 7, such rare mislabeled or semantically ambiguous samples become extremely difficult to infer, but the prior from the RL module could provide strong 'advice' on such outliers based from its dense knowledge spanning *entire* relationship space. Additional interesting successful/failed prediction results are in the supplement.

6. Conclusion

We presented a novel end-to-end pipeline for the visual relationship detection problem. We first exploits a simple tensorial representation of the training data and derives a powerful relational prior based on a algebraic formulation to obtain latent "factorial" representations from the sparse tensor via a novel spectral initialization. Our results suggest that the factors can be provably learned from observations only logarithmic in the number of relationships given the ill-posedness of the problem. With this regularization, we show how informing an end-to-end visual relationship detection pipeline with such a distilled prior yields state-of-the-art in predicate and scene graph predictions.

7. Acknowledgment

We acknowledge UW CIBM fellowship 5T15LM007359-14 (SJH), UW CPCP AI117924 (SNR), NSF CAREER award 1252725 (VS) and NSF CCF 1320755 (VS). Preceding work done while MDC was an intern at MSR, and we thank Pushmeet Kohli, Matthew Johnson and Maximilian Nickel for discussions on relational learning.

Code available on https://github.com/shwang54

References

[1] M. Alberti, J. Folkesson, and P. Jensfelt. Relational approaches for joint object classification and scene similarity measurement in indoor environments. In *AAAI 2014 Spring Symposia: Qualitative Representations for Robots*, 2014. 1

[2] A. Anandkumar, R. Ge, D. J. Hsu, S. M. Kakade, and M. Telgarsky. Tensor decompositions for learning latent variable models. *JMLR*, 15, 2014. 4

[3] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Deep compositional question answering with neural module networks. In *CVPR*, 2016. 3

[4] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. VQA: Visual Question Answering. In *ICCV*, 2015. 1, 2, 3

[5] Y. Atzmon, J. Berant, V. Kezami, A. Globerson, and G. Chechik. Learning to generalize to new compositions in image understanding. *arXiv preprint arXiv:1608.07639*, 2016. 2

[6] T. Bansal, C. Bhattacharyya, and R. Kannan. A provable SVD-based algorithm for learning topics in dominant admixture corpus. In *NIPS*, 2014. 5

[7] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, Belmont (Mass.), 1999. 6

[8] C. Boutsidis and E. Gallopoulos. SVD based initialization: A head start for nonnegative matrix factorization. *Pattern Recognition*, 41, 2008. 5

[9] A. Chandrasekaran, A. K. Vijayakumar, S. Antol, M. Bansal, D. Batra, C. Lawrence Zitnick, and D. Parikh. We are humor beings: Understanding and predicting visual humor. In *CVPR*, 2016. 1

[10] X. Chen and C. Lawrence Zitnick. Mind's eye: A recurrent visual representation for image caption generation. In *CVPR*, 2015. 3

[11] W. Choi, Y.-W. Chao, C. Pantofaru, and S. Savarese. Understanding indoor scenes using 3d geometric phrases. In *CVPR*, 2013.

[12] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-scale object classification using label relation graphs. In *ECCV*, 2014. 1

[13] C. Desai and D. Ramanan. Detecting actions, poses, and objects with relational phraselets. In *ECCV*, 2012. 1

[14] S. A. Eslami, N. Heess, T. Weber, Y. Tassa, D. Szepesvari, G. E. Hinton, et al. Attend, Infer, Repeat: Fast Scene Understanding with Generative Models. 2016. 2

[15] N. Friedman, L. Getoor, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In *IJCAI*, 1999. 3

[16] L. Getoor, N. Friedman, D. Koller, and B. Taskar. Learning probabilistic models of relational structure. In *ICML*, 2001. 3

[17] J. Gorski, F. Pfeuffer, and K. Klamroth. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 66, 2007. 6

[18] R. A. Harshman and M. E. Lundy. PARAFAC: Parallel factor analysis. *Computational Statistics & Data Analysis*, 18, 1994. 4, 6, 7

[19] D. Hsu and S. M. Kakade. Learning mixtures of spherical gaussians: moment methods and spectral decompositions. In *IICS*, 2013. 4

[20] P. Jain, R. Meka, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. In *NIPS*, 2010. 5

[21] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015. 1, 2, 6

[22] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009. 4

[23] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. In *IJCV*, 2016. 2, 3, 6

[24] J. S. Kulchandani and K. J. Dangarwala. Moving object detection: Review of recent research trends. In *ICPC*. IEEE, 2015. 2

[25] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*, 2010. 1, 3

[26] K. Lebeda, S. Hadfield, and R. Bowden. Exploring Causal Relationships in Visual Object Tracking. In *ICCV*, 2015. 2

[27] W. Li, J. Joo, H. Qi, and S.-C. Zhu. Joint Image-Text News Topic Detection and Tracking by Multimodal Topic And-Or Graph. *Multimedia*, 2016. 1

[28] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*. Springer, 2014. 2

[29] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016. 1, 2, 6, 7, 8

[30] C. Lu, J. Tang, S. Yan, and Z. Lin. Generalized nonconvex nonsmooth low-rank minimization. In *CVPR*, 2014. 5

[31] Y. Lu, T. Wu, and S. Chun Zhu. Online object tracking, learning and parsing with and-or graphs. In *CVPR*, 2014. 1

[32] T. Mensink, E. Gavves, and C. G. Snoek. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, 2014. 1, 3

[33] M. Nickel, V. Tresp, and H.-P. Kriegel. A three-way model for collective learning on multi-relational data. In *ICML*, 2011. 4

[34] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. 6

[35] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa. Visual domain adaptation: A survey of recent advances. *Signal Processing Magazine*, 2015. 2

[36] V. Ramanathan, C. Li, J. Deng, W. Han, Z. Li, K. Gu, Y. Song, S. Bengio, C. Rosenberg, and L. Fei-Fei. Learning semantic relationships for better action retrieval in images. In *CVPR*, 2015. 3

[37] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 6

[38] M. Richardson and P. Domingos. Markov logic networks. *Machine learning*, 62, 2006. 3

[39] M. Rohrbach, W. Qiu, I. Titov, et al. Translating video content to natural language descriptions. In *ICCV*, 2013. 3

[40] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *CVPR*, 2011. 1, 2, 3

[41] S. Sanghavi, R. Ward, and C. D. White. The local convexity of solving systems of quadratic equations. *Results in Mathematics*, pages 1–40. 5

[42] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *ACM SIGKDD*. ACM, 2008. 3

[43] Y. C. Song, H. Kautz, J. Allen, M. Swift, Y. Li, J. Luo, and C. Zhang. A markov logic framework for recognizing complex events from multimodal data. In *ICMI*. ACM, 2013. 1

[44] R. Sun and Z.-Q. Luo. Guaranteed matrix completion via nonconvex factorization. In *FOCS*, 2015. 5

[45] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. J. Mooney. Integrating Language and Vision to Generate Natural Language Descriptions of Videos in the Wild. In *COLING*, 2014. 3

[46] S. D. Tran and L. S. Davis. Event modeling and recognition using markov logic networks. In *ECCV*, 2008. 1

[47] S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht. Low-rank solutions of linear matrix equations via procrustes flow. *arXiv preprint arXiv:1507.03566*, 2015. 4, 5

[48] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31, 1966. 4, 6, 7

[49] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *CVPR*. IEEE, 1991. 4

[50] J. Wang, M. Korayem, S. Blanco, and D. J. Crandall. Tracking Natural Events through Social Media and Computer Vision. In *Multimedia*. ACM, 2016. 2

[51] Q. Wu, P. Wang, C. Shen, A. Dick, and A. van den Hengel. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *CVPR*, 2016. 1

[52] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017. 1, 2, 3, 6, 7, 8

[53] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *ICML*, 2015. 3

[54] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010. 1, 3

[55] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei. Visual7W: Grounded Question Answering in Images. In *CVPR*, 2016. 1, 2, 3

[56] C. L. Zitnick, R. Vedantam, and D. Parikh. Adopting abstract images for semantic scene understanding. *PAMI*, 2016. 2