Predicting Object Dynamics in Scenes

David F. Fouhey
Carnegie Mellon University
dfouhey@cs.cmu.edu

C. Lawrence Zitnick
Microsoft Research
larryz@microsoft.com

Abstract

Given a static scene, a human can trivially enumerate the myriad of things that can happen next and characterize the relative likelihood of each. In the process, we make use of enormous amounts of commonsense knowledge about how the world works. In this paper, we investigate learning this commonsense knowledge from data. To overcome a lack of densely annotated spatiotemporal data, we learn from sequences of abstract images gathered using crowdsourcing. The abstract scenes provide both object location and attribute information. We demonstrate qualitatively and quantitatively that our models produce plausible scene predictions on both the abstract images, as well as natural images taken from the Internet.

1. Introduction

What can happen next in Figure 1? Where should the sunglasses in the middle of the image go? Humans can readily distinguish plausible continuations of Figure 1's scene from nonsense ones: the people will probably move forward, the woman on the left may stop smiling, the sunglasses will likely remain attached to the people's faces, and if there were sunglasses on the street they would probably remain there. Despite its complexity, prediction is as effortless for humans as recognizing the contents of the image. In this paper, we investigate how we can represent and predict the dynamic high-level aspects of a scene.

How can we model the dynamics of objects? Several factors are critical in motion prediction. These include the likelihood of the object's resulting position. For instance, a person is more likely to be standing on a street than in the air. Similarly, the likelihood of the object's motion needs to be considered. For instance, a dropped ball will fall towards the ground and not away from it. A third and more subtle factor also must be considered. The previous two factors examine the movement of an object independent of the movement of other objects. However, the movement of several objects may be strongly dependent. For example, if two objects are attached to each other, they will move together, such as a pair of sunglasses worn by a person. Finally, the changes that occur within a scene are not just limited to the movement of objects. Animate objects can change in pose, gaze direction and facial expression. These object attributes are essential for both predicting scene motions and for generating realistic predictions of future attributes.

Figure 1. Given an image, many things are perceived to be likely to happen next and many things are not. Much of what separates the plausible from the implausible is commonsense spatiotemporal knowledge. Irrespective of what happens next, we know that the woman's sunglasses should move with her and that the streetlight should stay in place. In this paper, we explore how to learn commonsense spatiotemporal knowledge from data.

How can we model these various factors? A model must capture the dependence of an object's prediction on the previous objects' attributes and positions, while enforcing constraints on the predicted relative motions of objects. In this paper, we achieve these goals using a Conditional Random Field (CRF) formulation. The nodes of the CRF represent the objects and their attributes, while pairwise potentials model their interactions. The unary and pairwise potentials are learned using a Random Forest classifier with the previous objects' positions and attributes as features. The pairwise potentials enable constraints to be placed on the relative motion between pairs of objects. A multiple-restart iterative conditional modes algorithm is used to sample novel scenes from the model.

An enormous challenge for studying the prediction of scene dynamics is obtaining training data. As motivated by Fig. 1, we need extraordinarily detailed knowledge of

the scene if we are to make effective predictions. Getting a reasonable amount of non-contrived image sequence data adequately annotated is prohibitively difficult. But more importantly, even if we could get sufficient training data, at inference-time we would need fairly accurate answers to questions that are far too difficult for contemporary computer vision algorithms, e.g., detecting purses and sunglasses, or localizing hands in unconstrained images.

Deferring the investigation of scene dynamics until better detectors are developed is short-sighted. Incorporating an understanding of scene dynamics ought to provide useful complementary information for detecting objects and related tasks in image sequences. Nonetheless, simply developing prediction on top of contemporary vision algorithms is a risky endeavor, since it may be unclear whether deficiencies are due to failings in object detection or in prediction. We therefore turn to abstract scenes [21, 22], which allow us to investigate scene dynamics in isolation and to easily gather large amounts of data. This methodology allows us to easily test multiple models to learn which may be most promising when applied to real images. Finally, while our models are learned on abstract scenes, we demonstrate their surprising ability to generalize to natural images.

We show that our model outperforms a number of alternate baseline approaches to modeling scene dynamics on both abstract and natural images. We report quantitative results using human studies, as well as measuring results on various prediction sub-tasks.

2. Related Work

Learning spatiotemporal commonsense knowledge for predicting scene dynamics offers many un-explored challenges. Most work on reasoning about future actions has used top-down and manually encoded commonsense knowledge. In the tracking literature, this has been done by treating humans as privileged objects and reasoning about their motion around defined obstacles [6] and other pedestrians [14, 15]. Analogously in the affordance literature [8], common sense takes the form of geometric knowledge of where humans can perform various actions. In contrast, we investigate the problem of learning these relationships themselves, including the fact that animate objects, especially humans, are special objects with respect to motion.

Other work has aimed at learning these relations from data. For instance, Lan et al. [12] learn social relationships from video data that are informative of human actions. Kitani et al. [10] learn which parts of scenes humans should prefer while moving. Xie et al. [18] learn functional parts of scenes for predicting human paths, and Koppula et al. [11] learn to predict human actions given the presence of other objects. However, significant complexity exists beyond human motion. Inanimate objects are frequently in motion, e.g., if they are already in motion or attached to an actor. It is not clear how to learn that humans are special with respect to other objects and how this relationship should be modeled.

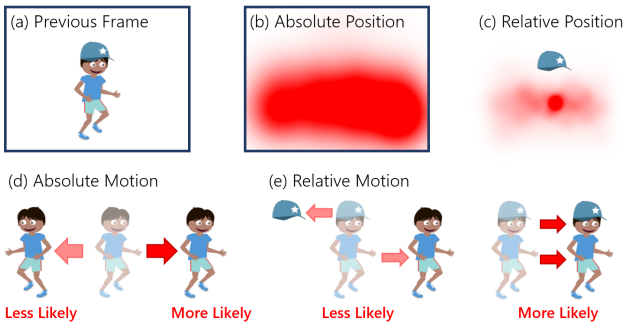


Figure 2. Illustration of the four factors modeled in our motion prediction model: (a) Previous frame at time $t-1$ for a boy and hat, (b) the absolute position at time t for the boy, (c) the relative position for the boy and the cap at time t, (d) the absolute motion of the boy, and (e) the relative motion of the boy and the cap. Notice it is more likely for the hat to move with the boy than not.

This difficulty has also noted in recent work [17] that, in addition to prediction, tackles the challenge of learning which parts of the scene will move without assuming a-priori semantic knowledge. Nonetheless, as confirmed by our experiments, the motions of objects are highly interdependent and thus modeling them independently produces implausible results. One work that does not follow this paradigm is that of Yuen and Torralba [20], which models each scene as a whole and transfers scene-level flow vectors. While this yields plausible flow fields for test scenes, the more difficult question of how these fields can be combined to produce a following scene remains unanswered. As shown by Fig. 2(e) and our baseline results using an approach with global motion transfer, holistically applied scene flow can lead to poor results.

There is a rich body of literature exploring spatial commonsense. This comes in the form of context models for improving object recognition [5, 9], recognizing out-of-place objects [4], as well as learning spatial relationships [7]. While much of this work is top-down, there has been recent work [3] proposing the automatic learning of visual commonsense from Internet images. Nonetheless, spatiotemporal reasoning is not a simple extension of spatial reasoning: the heads of people tend to be likely locations of hats, but when predicting the motion of a group of people and their hats, the hats' locations cannot be arbitrarily permuted.

Artificial intelligence has also focused on obtaining commonsense knowledge such as properties and relationships among concepts via automated reasoning [13] and by reading the Web [1]. These systems include many of the same ingredients as predicting scene dynamics, including predicting how things change in time [16]. Unfortunately, this commonsense is not visually-grounded and is entirely abstract. At best, it might suggest that if a human is carrying a box, the box will move with the human; however, it cannot reveal how to recognize that a human is carrying a box,

even if one can recognize both humans and boxes.

Our work specifically tackles the problem of learning the dynamic structure of scenes. This requires joint reasoning about objects in both space and time to learn commonsense relationships. In our work, we automatically learn under what conditions objects are capable of motion, how they move, and the interdependence of object motion. Unlike much past work, we do not assume a priori semantic knowledge of objects, but instead model each object equivalently and learn the its properties from data.

Recently, several works have explored semantic scene understanding using abstract scenes similar to ours [21, 22]. The semantic understanding of dynamic scenes also requires temporal commonsense knowledge and has also recently gained attention [2, 19].

3. Overview

Prediction is an inherently ambiguous problem. A myriad of events may follow from a particular scene with varying likelihoods. Given a set of previous scenes and model parameters, our approach computes the likelihood of a predicted scene. During generation, we sample from this model to predict scenes of high likelihood. This allows our model to acknowledge the multi-modal nature of prediction.

We model four aspects of scene composition and transition: the position and motion of each object in either absolute terms or relative to other objects (Fig. 2). Our model captures the rules of scene composition with constraints on both the absolute location of objects (e.g., a boy is likely to be standing on the ground), as well as the relative location of objects, e.g., a boy is likely to be below a baseball cap (Fig. 2(c)). To capture temporal information, we learn a model of the absolute motion of the objects, e.g., a child is more likely to move in the direction they are facing (Fig. 2(d)). Finally, we model the relative motion of pairs of objects in scene transitions. This subtle yet important factor ensures that objects move coherently. For instance, as illustrated in Fig. 2(e), it is more likely for the boy and the baseball cap to move in the same direction than not.

3.1. Spatiotemporal Model

We begin by describing how the various spatial and temporal factors are represented using a Conditional Random Field (CRF). Our approach for sampling and generating scenes from the model is described in the following sections. Each node in our CRF represents an object and the edges model the relationships between objects. Our dataset consists of 58 different objects [21] that are commonly found outside when children are playing, such as people, toys, clothing, animals, trees, etc. Section 4 describes how our training data set was creating using clip art and Amazon's Mechanical Turk. For testing, we show results on both abstract and real scenes.

Each object is represented as a node in the CRF with the following variables: location $\{x_i, y_i\}$, depth represented as the scale s_i of the object, and the horizontal orientation or flip $d_i \in \{-1, 1\}$, $\Phi_i = \{x_i, y_i, s_i, d_i\}$. Each object may also have a set of attributes Ψ_i. Currently, only people have attributes corresponding to their pose and facial expression. However, the model is general and could handle attributes for other objects as well. Finally, we denote the previous set of scenes with S and the model parameters as Θ. Our model is of the form:

$$\begin{aligned}\log P(\Phi, \Psi | S, \Theta) = \\ \sum_i \left(\overbrace{\lambda(\Phi_i; \theta_\lambda)}^{abs.\ location} + \overbrace{\omega_i(\Phi_i, S; \theta_\omega)}^{abs.\ motion} + \overbrace{\pi_i(\Psi_i, S; \theta_\pi)}^{attributes} \right) \\ + \sum_{i,j} \left(\overbrace{\phi_{i,j}(\Phi_i, \Phi_j; \theta_\phi)}^{rel.\ location} + \overbrace{\varphi_{i,j}(\Phi_i, \Phi_j, S; \theta_\varphi)}^{rel.\ motion} \right) \\ + \overbrace{\alpha(\Phi, S; \Theta_\alpha)}^{motion\ prior} - \log Z(S, \Theta) \end{aligned} \quad (1)$$

where λ, ω, π are unary potentials, ϕ, φ are binary potentials, α is a global potential, and $Z(S, \Theta)$ is the partition function that normalizes the distribution. The variables i and j index objects. Next, we describe how each potential is computed, followed by describing the method used to learn the model parameters that are dependent on the previous scenes.

Absolute location: The absolute location of an object in a scene is modeled using a spherical Gaussian Mixture Model (GMM):

$$\lambda(\Phi_i; \theta_\lambda) = \log \left(\sum_k P(\Phi_i | k) \theta_\lambda(i, k) \right)$$

where k indexes the GMM's components. $P(\Phi_i|k) = \mathcal{N}((x_i, y_i); \mu_k, \sigma_k)$ is the normal distribution with mean μ_k and standard deviation σ_k. In our experiments, 40 components are used. The means μ_k and standard deviations σ_k are jointly learned across all object types. The mixture coefficients $\theta_\lambda(i, k)$ for the ith object and kth component are set equal to the empirical likelihood $P(k|i)$.

Absolute motion: Absolute motion represents the independent motion of an object. Similar to absolute location, the absolute motion potential is computed using a GMM,

$$\omega_i(\Phi_i, S; \theta_\omega) = \log \left(\sum_k P(\delta(\Phi_i, \Phi_i') | k) \theta_\omega(i, k, S) \right)$$

where $P(\delta(\Phi_i, \Phi_i')|k)$ is the Probability Density Function (PDF) of the kth Gaussian component of the GMM learned for absolute motion. Similarly to the absolute location GMM the means and standard deviations are shared across objects. The distance between the object at its predicted

location Φ_i and its previous location Φ'_i is computed by adjusting for horizontal orientation d_i and scale s_i,

$$\delta(\Phi_i, \Phi'_i) = \left(\frac{d_i(x_i - x'_i)}{s_i}, \frac{(y_i - y'_i)}{s_i} \right), \quad (2)$$

where $\{x'_i, y'_i\} \subset \Phi'_i$ is the position of the object in the previous time frame. We describe how we learn the model parameters $\theta_\omega(i, k, S)$ in the subsequent section.

Attributes: The attribute potential represents the likelihood of seeing each attribute in the next scene given the previous scene,

$$\pi_i(\Psi_i|\theta_\pi) = \log \left(\sum_k \theta_\pi(i, k, S) \right),$$

where k indexes a set of attributes. We describe how we obtain the parameters $\theta_\pi(i, k, S)$ in the subsequent section.

Relative location: Relative location represents the likelihood of seeing object j in a particular position relative to object i, and is also represented by a GMM:

$$\phi_{i,j}(\Phi_i, \Phi_j|\theta_\phi) = \log \left(\sum_k P(\delta(\Phi_i, \Phi_j)|k)\theta_\phi(i, j, k) \right).$$

As with the GMM used to model absolute position, the component PDFs are learned jointly across all objects. The parameters $\theta_\phi(i, j, k)$ are set to the empirical likelihood $P(k|i, j)$ of the kth component given object i and j in the training data.

Relative motion: Relative motion represents the likelihood of one object j's motion vector relative to another object's, and is factored as:

$$\varphi_{i,j}(\Phi_i, \Phi_j, S|\theta_\varphi) = \log \left(\textstyle\sum_k P(\varepsilon(\Phi_i, \Phi_j, S)|k)\theta_\varphi(i, j, k, S) \right),$$

where ε measures the difference in motion vectors similar to Equation 2 using,

$$\varepsilon(\Phi_i, \Phi_j, S) = \left(\frac{d_i(x_i - x'_i) - (x_j - x'_j)}{s_i}, \frac{(y_i - y'_i) - (y_j - y'_j)}{s_i} \right). \quad (3)$$

We describe how we obtain the model parameters $\theta_\varphi(i, j, k, S)$ in the subsequent section.

Motion Prior: Our final potential places a prior on the number of objects that move in a scene. Without such a prior, the most likely prediction is often that no objects move. To preclude this from happening we add a motion prior on the number of objects that moved M,

$$\alpha(\Phi, S) = \log \theta_\alpha(M),$$

where θ_α is the empirical frequency of observing M moving objects.

3.2. Learning model parameters

Many of our scene potentials are dependent on the state of the previous scenes, S. In our paper S either contains one or two previous scenes. The parameters related to the absolute or relative movement of an object are dependent on S. For instance, hats and people move together only if they are attached, and a hot dog is more likely to move near a human than by itself. Changes in attributes are also dependent on previous scenes. For instance, if a bear moves towards a person their expression might change to fear. In this section, we describe how the model parameters for the temporally dependent potentials in our CRF are computed. These include the absolute motion θ_ω, relative motion θ_φ, and attribute θ_π model parameters.

To estimate our parameters, we learn a mapping from S. This mapping consists of two stages. First, a classifier determines whether an object i will move $\Omega(i, S) \in \{0, 1\}$, and another classifier $\Pi(i, S) \in \{0, 1\}$ determines if its attributes will change. If no change occurs, the model parameters are held constant from the previous scene. Otherwise, the model parameters are updated using a series of classifiers Υ for each set of parameters and object. For instance, a classifier $\Upsilon_\omega(i, S)$ is learned that takes as input features extracted from the previous scenes S related to object i and outputs a k dimensional feature vector corresponding to the mixture coefficients used by the absolute motion GMM, $\theta_\omega(i, k, S) = \Upsilon_{\omega,k}(i, S)$, with $\sum_k \Upsilon_{\omega,k}(i, S) = 1$. Similar classifier pairs are learned for relative motion and sets of exclusive attributes, such as poses and expressions; pairwise classifiers are learned bi-directionally (i.e., the hat's motion relative to the girl and the girl's motion relative to the hat). Each classifier is learned using multi-class random forests with k classes using the Gini coefficient for splitting. For training, the values of $\theta_\omega(i, k, S)$ and $\theta_\varphi(i, j, k, S)$ are computed from the response of their respective Gaussian components on the ground truth next scene. The attribute parameters are set to the observed attributes θ_π in the ground truth scene. Only objects that change position or attributes are used for training Υ. The sets of scenes used for training are described in Section 4.

Features: Each of our classifiers Ω, Π and Υ compute a set of features from the previous scenes S that are used as inputs into a random forest classifier. For every object's classifier, we use the absolute location of the object as well as the relative location of the other objects encoded by the response of the Gaussian components in the relative location GMMs. If an object does not exist in a scene, all of its GMM features are zero. The attributes for each object are encoded as a set of binary values. The GMM features and attribute features for all objects are concatenated to form our final feature vector. Given 58 objects, 40 mixture components per GMM, and 11 attribute values, our feature vector has size $40+40\times57+11\times58 = 2958$. If S has two scenes, we concatenate the features from the most recent scene with a motion feature, specifically the object's motion between

the two scenes encoded with the absolute motion GMM as well as whether each attribute changed.

3.3. Prediction generation

We generate a predicted scene in two stages. First, using Ω and Π, we generate a posterior on which objects will move and/or have their attributes changed. For those objects which do change, we estimate new parameters θ_ω, θ_φ and θ_π using Υ_ω, Υ_φ and Υ_π respectively. Given the estimated set of parameters for the updated CRF, we use a form of Iterated Conditional Modes (ICM) to generate a scene. We randomly select an object that is changing position and estimate its most likely position assuming the other objects' positions are fixed, and similarly for attributes. After a fixed number of iterations, the scene is scored using the CRF. This procedure is repeated 2,000 times and the scenes with the best scores are returned.

3.4. Implementation details

Number of trees: for k-way classification problems, we use 100 trees; for distributions, we use 30 trees. These and other parameters were capped to ensure training and testing could be done within 16 hours on a single desktop. **Pose alignment:** To determine whether an object stays attached to a person's head or hand even if their pose changes, we assume that we have correspondence between poses. We manually annotate the discrete poses of humans with landmarks and compute the relative position of nearby objects with respect to these landmarks. We warp the relative position of nearby objects (within a 20% bounding-box size radius) to a canonical reference frame. **Learning the mixture models:** We learn the four GMMs used for inference using k-means, yielding spherical components. **Robustness to slight misalignment:** To give robustness to the tool used to create scenes, which does not naturally allow users to move objects together, we clip motions computed to be less than 30 pixels (normalized for scale) to 0 in our learning and inference.

4. Results

We now present experiments to validate our approach. In addition to the qualitative results throughout the paper, we present a quantitative analysis using human studies as well as results on various prediction sub-tasks, such as determining whether an object should move. Finally, we present results on natural photos gathered from the Internet using similar bounding-box-style annotations.

Abstract scene dataset: We gathered a dataset of $5,000$ sequences from Amazon Mechanical Turk (AMT). Turkers were asked to create a story consisting of five frames. Each worker had access to a random subset of items from [21] and could arbitrarily arrange objects and set them to three discrete scales and two horizontal orientations. As Turkers moved forward in time, the interface propagated the previous scene. We split the data randomly into $4,500$ training and 500 testing sequences. An example of the user interface may be viewed in the supplementary material.

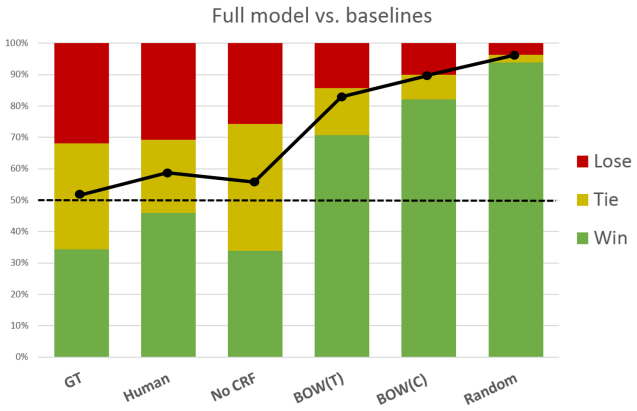


Figure 3. Human evaluation of the Full CRF predictions compared to the baselines (win, tie, lose.) The full method substantially outperforms the computer baselines and even crowdsourced human completions while doing comparably to the ground-truth scenes. The black line shows the win/loss ratio for each baseline.

Methods: We compare our Full CRF approach with five baselines. The first two provide a sense of inter-human agreement; the next two test whether it is possible to simply transfer an entire scene; the last assesses whether prediction may be handled independently for each object. **(1) Ground-truth:** Given a test scene, we return the following scene in the sequence. **(2) Human prediction:** We crowdsource the prediction by asking a human to continue the scene. We show subjects the first two scenes and ask them to complete the sequence. **(3) Bag-of-Words copy (BoW(C)):** Given the test scene, we build a multiscale bag-of-words representation on object presence and find the most similar scene in our training dataset. We use the scene that follows it as the prediction. When given multiple scenes, we concatenate the representations. **(4) Bag-of-Words transfer (BoW(T)):** Simply copying the next scene makes no adjustments for scale or position. We instead transfer the motions of the objects common to both the query and retrieval scene, adjusting for scale and orientation. **(5) No CRF:** We train a random forest to predict the location and attributes of an object in the scene using the same features as our full method; we take the most likely label for all locations and attributes. This does not model the relative motion or enforce consistency among the objects' predictions as done by the full CRF. **(6) Full CRF:** This is our full model including the CRF.

4.1. User study results

One natural way to evaluate predictions is asking humans to assess the likelihood of one scene following another. To do this, we ask human subjects to judge the relative quality

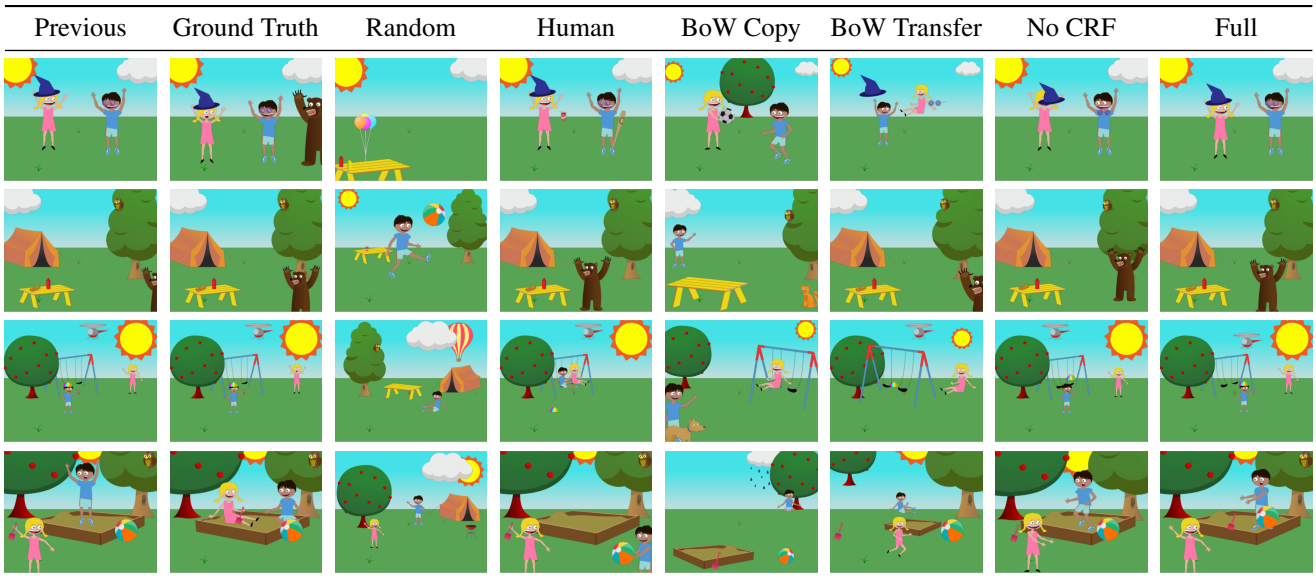


Figure 4. The proposed approach compared with the baselines on scenes selected from ones with consistent human ratings (top three rows success, final row failure). Simply copying the scene results in continuity errors (BOW Copy, Random), e.g., unrealistic disappearance of objects. Transferring the motions (BOW Transfer) or predicting without a CRF (No CRF) does not enforce consistency in scene dynamics (note the inaccurate motion of hat and sunglasses in the top row).

of the approaches. We show subjects the first two scenes of a sequence and then the completions generated by two approaches. The subjects can pick either scene to be more likely or declare a tie. We compare our full approach with each baseline in Fig. 3. Qualitative results selected based on these evaluations are shown in Fig. 4.

Our method is comparable to the ground truth and substantially out-performs the other algorithmic approaches. The bag-of-words approaches do especially poorly in comparison. Transferring the motion rather than simply returning the next scene results in a modest, but ultimately inadequate improvement because it is hard to find similar scenes with similar layouts given the huge diversity of scenes. As a result, object motions typically appear unnatural with attachment properties frequently violated, e.g., the third row of Fig. 4.

We also outperform the No CRF approach that does not model relative motion and other joint factors handled by our Full CRF approach. As shown in Fig. 4, this may produce inaccurate predictions, causing our full method to be preferred 34% of the time and the No CRF being preferred 26% of the time. Since modeling joint factors is not critical for all scenes, the two approaches are found to tie in 40% of instances. The crowdsourced human completions are often overly creative, resulting in worse results.

4.2. Quantitative results

Another approach to evaluating prediction is evaluating the extent to which a prediction agrees with what actually happened in a sequence of scenes. In aggregate, a good predictor will correlate better with the actual continuation of a scene than a bad one.

We evaluate various prediction sub-tasks quantitatively using standard metrics. Specifically, we use precision (whether the predictor is right when it predicts the positive class), and recall (whether ground-truth positives are predicted as positive). We summarize precision and recall with the F1 score, or harmonic mean. For attributes, we compute F1 scores for each attribute and take their average.

Metrics: We now describe and motivate each sub-task used. **Motion Indicator:** If an object is present at both time t and $t+1$, does it move from time t to $t+1$? This metric captures both objects that should not move (e.g., trees) and objects whose motion is scene-dependent (e.g., hot dogs). **Joint Motion:** If object i moves from time t to $t+1$, which objects have the same displacement as object i? This metric characterizes how well a model captures interactions between objects. Even if a method accurately models the objects' individual motion, its predictions will be incoherent if it does not capture the relationships between the objects' motions. **Motion Direction:** If an object moves, does it move left or right? **Attributes:** If an object is present at both time t and $t+1$, what are its attributes at time $t+1$?

Results: We present quantitative results in Table 1 for all methods. We independently show results with one scene and two scenes as input; our approach outperforms the baselines according to most criteria. The F1 measure for the motion indicator and attributes is substantially better for Full CRF and No CRF since both use the same learned functions $\Omega(i, S)$ and $\Upsilon_\pi(i, S)$ to determine whether an object moves and how its attributes change respectively. Note that recall is higher for BoW Copy and Random since they al-

Table 1. Quantitative evaluation of the approach in comparison to the baselines on the abstract scenes dataset. We report results with models using both one and two previous scenes.

		One Previous Scene						Two Previous Scenes			
		Full CRF	Human	BoW(C)	BoW(T)	No CRF	Random	Full CRF	BoW(C)	BoW(T)	No CRF
Motion Indicator	F1	**49.3**	48.8	33.9	39.6	49.0	39.0	**53.3**	33.3	36.4	50.4
	Prec.	**51.2**	50.1	20.9	42.8	46.1	24.5	**56.0**	20.3	40.7	50.3
	Rec.	47.6	47.5	87.7	36.9	52.4	**96.0**	50.9	**91.4**	32.9	50.6
Joint Motion	F1	**42.9**	31.8	16.8	16.2	11.9	0.0	**44.9**	18.6	10.0	15.6
	Prec.	**42.9**	31.3	13.6	20.0	9.0	0.0	**46.8**	17.4	25.0	12.3
	Rec.	**42.9**	32.3	18.8	13.6	17.7	0.0	**43.1**	20.0	6.3	21.2
Motion Direction	F1	66.6	70.0	57.7	61.9	**75.1**	65.8	68.7	63.0	63.1	**74.4**
Attributes	F1	**61.9**	52.3	24.6	29.1	61.5	22.1	**61.9**	23.9	27.9	61.6

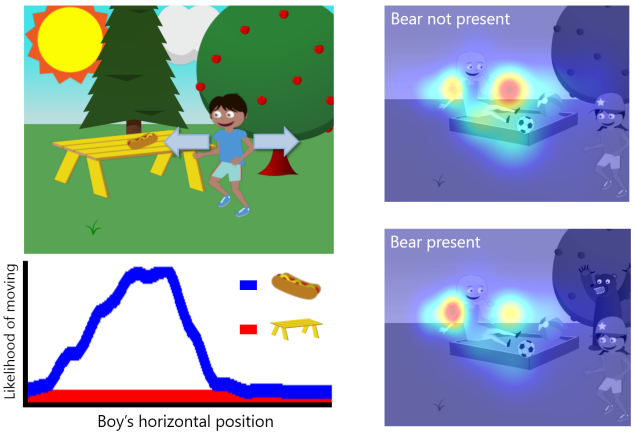


Figure 5. How do interactions influence predicted motion? **(Left)** We plot the relative likelihood Ω that an object will move as we sweep the boy across the scene for the hot dog (blue) and picnic table (red). Notice the hot dog is more likely to move if the boy is near it. **(Right)** we show the distribution (red more likely; blue less likely) of the girl's absolute motion Υ_ω in one scene (top) and if we just add a bear (bottom). Although there is a strong prior for moving forward, the bear's presence overrides this.

most always predict objects will move. For joint motion, Full CRF significantly outperforms No CRF and the other baselines. The F1 measure is over twice as good as the next best non-human baseline method. This shows that joint object dynamics do not emerge from per-object motion models in the No CRF method and are similarly not captured by non-parametric transfer via either Bag-of-Words models. This can be seen qualitatively in Fig. 4, rows 1 and 3.

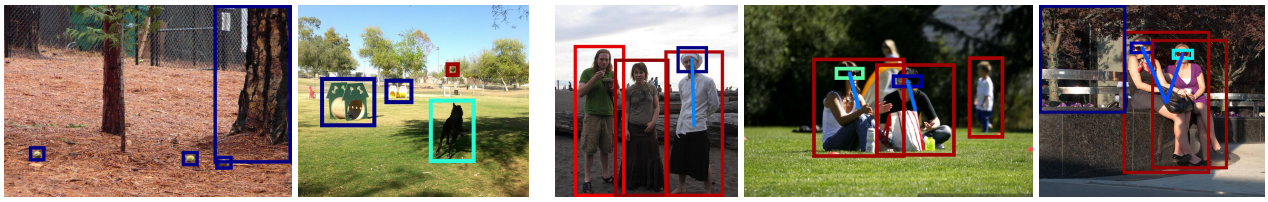
The No CRF approach achieves the best result on predicting the motion direction. We found the No CRF approach heavily favors moving objects to the center, which is commonly correct. The high value for the Random predictor can be explained by the same effect in that it typically pulls objects towards the center. The pairwise relative motion constraints can sometimes pull objects away from the center, resulting in slightly lower scores. However, as shown in Fig. 3 and the joint motion metric, when the directions and magnitudes of the movements are considered together, the Full CRF produces more convincing results.

Two qualitative results are shown in Fig. 5 demonstrating the effect of other objects on an object's predicted motion. For instance, if the boy is near a hot dog, the hot dog is more likely to move, unlike a stationary object like a picnic table. The presence of objects such as bears can also effect the likelihood of an object's motion direction.

4.3. Natural Images

We also evaluate our approach on natural images. We gathered 225 images from `flickr.com` using queries containing descriptions of one or more pieces of clip art (e.g., "dog tennis ball"). The bounding boxes of object categories that exist in our clip art dataset were labeled using AMT. Because we do not know what occurred in the scene after the image was taken, we rely on human annotation for ground truth labels on our prediction sub-tasks. We apply each model learned on abstract scenes and compare with our ground-truth labels for whether objects move and which objects move together. For human results on natural images, we ask a separate set of annotators to label the ground-truth and treat these as predictions; these results are generally better than our human baseline on abstract scenes since humans do not have to actually produce a scene, but instead assess how the objects will move.

As shown in Table 2, we outperform our baseline approaches on real images. As with abstract images, the F1 measure for joint motions in natural images is significantly better than the computer baselines. Our results are similar to the human baseline for the motion indicator metric. However the humans have access to richer information and predict joint motions better. We show qualitative results in Figs. 6 and 7. Fig. 6(a) shows that our model can learn subtle rules, such as balls are unlikely to move unless they are in the air or near an agent that can move them. Similarly, Fig. 6(b) shows that our model's prediction of which objects move together captures notions of attachment, and Fig. 7 shows that our model can produce intuitively correct attributes.



(a) Context-dependent movement (b) Motion association

Figure 6. Results on our natural image dataset (best viewed in color): object with bounding boxes with warmer colors are more likely to move and lines join objects that are likely to move together. (a) The motion of many objects depends entirely on context: on the left, the tennis balls on the ground are unlikely to move; on the right, the ball flying through the air near the dog is very likely to move. (b) Our model captures motion association well, although it does not always determine attachment correctly for nearby humans.

Table 2. Quantitative evaluation of the approach in comparison to the baselines on the natural images dataset.

		Full CRF	Human	BoW(C)	BoW(T)	No CRF
Motion Indicator	F1	91.2	**97.8**	80.6	69.3	87.5
	Prec.	87.3	**95.6**	74.7	84.7	80.1
	Rec.	95.6	**100**	87.6	58.7	96.5
Joint Motion	F1	45.2	**75.3**	7.5	5.7	14.4
	Prec.	35.1	**60.4**	9.8	6.9	19.4
	Rec.	63.6	**100**	6.1	4.8	11.5

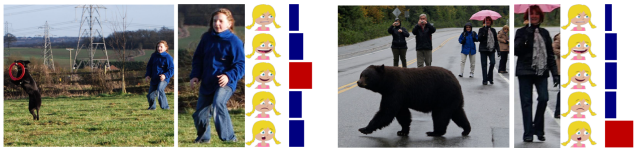


Figure 7. The model's belief Υ_π of peoples' expressions. In scenes with a bear, people are predicted as more likely to be scared.

5. Discussion

We have presented a method for modeling spatiotemporal dynamics for prediction in both abstract and natural scenes. Our method is capable of both sampling and evaluating the likelihood of future scenes and achieves substantially better performance compared to alternate approaches, including modeling at a scene level (BoW) and independently modeling each object (No CRF).

In the process, we have also offered insights into both the prediction task and its evaluation. Our results suggest that relative motion consistency does not simply emerge from the data, but must be built into a model to occur: both the global BoW models and the independent No CRF object approaches have dismal performance on joint motion metrics. This failure is reflected in both our quantitative metrics and human assessment.

In this paper, we explored short-range scene dynamics that occur within a brief time period. A promising area of future work is long-range interactions that commonly occur in coherent stories containing actors with specific personality types and comprehensive memories. How to semantically describe scene dynamics is also an open question.

References

[1] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. H. Jr., and T. M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI 2010*, 2010. 2

[2] D. L. Chen and R. J. Mooney. Learning to sportscast: a test of grounded language acquisition. In *ICML*. ACM, 2008. 3

[3] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. In *ICCV*, 2013. 2

[4] M. Choi, A. Torralba, and A. Willsky. Context models and out-of-context objects. *Pattern Recognition Letters*, 33(7), 2012. 2

[5] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert. An empirical study of context in object detection. In *CVPR*, 2009. 2

[6] H. Gong, J. Sim, M. Likhachev, and J. Shi. Multi-hypothesis motion planning for visual object tracking. In *ICCV*, 2011. 2

[7] A. Gupta and L. S. Davis. Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In *ECCV*, 2008. 2

[8] A. Gupta, S. Satkin, A. Efros, and M. Hebert. From 3d scene geometry to human workspace. In *CVPR*, 2011. 2

[9] D. Hoiem, A. Efros, and M. Hebert. Putting objects in perspective. *IJCV*, 80(1), 2008. 2

[10] K. M. Kitani, B. D. Ziebart, D. Bagnell, and M. Hebert. Activity forecasting. In *ECCV*, 2013. 2

[11] H. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. In *RSS*, 2013. 2

[12] T. Lan, L. Sigal, and G. Mori. Social roles in hierarchical models for human activity recognition. In *CVPR*, 2012. 2

[13] D. B. Lenat, R. V. Guha, K. Pittman, D. Pratt, and M. Shepherd. Cyc: toward programs with common sense. *Communications of the ACM*, 33(8), 1990. 2

[14] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force mode. In *CVPR*, 2009. 2

[15] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool. Youll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV*, 2009. 2

[16] P. P. Talukdar, D. Wijaya, and T. Mitchell. Acquiring temporal constraints between relations. In *CIKM*, 2012. 2

[17] J. C. Walker, A. Gupta, and M. Hebert. Patch to the future: Unsupervised visual prediction. In *CVPR*, 2014. 2

[18] D. Xie, S. Todorovic, and S.-C. Zhu. Inferring dark matter and dark energy from videos. In *ICCV*, 2013. 2

[19] H. Yu and J. M. Siskind. Grounded language learning from videos described with sentences. In *Association for Computational Linguistics*, 2013. 3

[20] J. Yuen and A. Torralba. A data-driven approach for event prediction. In *ECCV*, 2010. 2

[21] C. Zitnick and D. Parikh. Bringing semantics into focus using visual abstraction. In *CVPR*, 2013. 2, 3, 5

[22] C. Zitnick and D. Parikh. Learning the visual interpretation of sentences. In *ICCV*, 2013. 2, 3