Unsupervised Learning of Object Features from Video Sequences

Marius Leordeanu

Robert Collins

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

We develop an efficient algorithm for unsupervised learning of object models as constellations of features, from low resolution video sequences. The input images typically contain single or multiple objects that change in pose, scale and degree of occlusion. Also, the objects can move significantly between consecutive frames. The content of an input sequence is unlabeled so the learner has to cluster the data based on the data's implicit coherence over time and space. Our approach takes advantage of the dependent pairwise co-occurrences of objects' features within local neighborhoods vs. the independent behavior of unrelated features. We couple or decouple pairs of features based on a probabilistic interpretation of their pairwise statistics and then extract objects as connected components of features.

1. Introduction

Despite a lot of recent interest, learning from unlabelled data still remains one of the most challenging problems in the fields of computer vision and machine learning. Here we present an efficient method for learning object models as constellations of features in an unsupervised manner, from image sequences sampled from low resolution video.

Fergus et al ([1]) have proposed a method for unsupervised learning of object categories by fitting specific models to an input sequence containing objects of the same type, shown at different scales, similar pose, with cluttered backgrounds and a limited amount of occlusion. They model differently the object category, the occlusions and the background. They implicitly assume that the category of interest is the only coherent collection of parts over the input sequence, while the objects in the background display lack of structure and consistency. This assumption is valid so long as the object of interest is always in the foreground and covers a large part of the image, which requires careful selection of image sequences.

In video sequences the object of interest is not always in the foreground. New objects might come in the foreground and display consistency and coherence over time. We want to model the data in a way that will implicitly separate these objects. This is achieved by modelling the relationships between parts belonging to different objects (either background or foreground objects) as having random behavior, while assuming that parts belonging to the same object have consistent, structured behavior (Figure 1). These relationships between pairs of parts are independent of the camera position in the world or which is the object we are interested in. One application of this idea is in learning from tracking sequences, where the tracker might accidentally drift to a different object. Since our approach does not assume anything about which object/objects are present in each frame or their location in the image, we are able to separate the original object from the distracting objects.

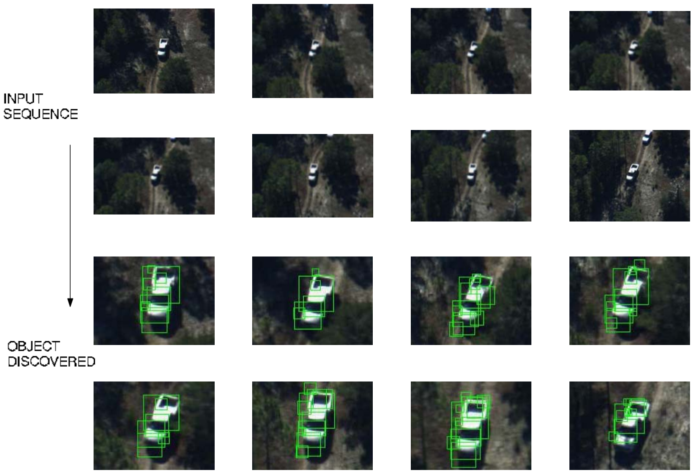


Figure 1: First two rows: subset of a 6 seconds long input sequence (frames were sampled at 5 fps). Last two rows: the object discovered (magnified). The squares displayed represent a connected component of co-occurring parts, classified as belonging to the same object

Ramanan and Forsyth pursue the idea of learning from tracking ([7]). They build models of animals from time coherent clusters that represent different body parts. Sivic and Zisserman attack the problem of extracting significant regions/patterns from video in [9]. Similar to data mining techniques for text databases, they extract neighborhoods of features that repeatedly reoccur over time. These papers do not explicitly model temporal dependencies between fea-

1063-6919/05 $20.00 © 2005 IEEE

tures, which leaves open the question of how to couple (decouple) parts that belong to the same (different) objects in a principled way.

In the case of rigid bodies and high resolution images, structure from motion techniques can be applied to recover the object models ([8]). However, in unsupervised scenarios, even more complex geometric techniques would need to answer the following question: for how long does a set of features have to display geometric consistency until it can be reliably classified as belonging to the same object ?

If their parts can be well detected and matched, even objects with a complex non rigid structure can be separated from the rest of the world by means of simple local co-occurrences of their features. While co-occurrences are relatively easy to monitor as compared to more complex geometric relationships, they are also very discriminative. This happens because features belonging to independent objects will not co-occur consistently, while features belonging to the same object will almost always co-occur. After recovering the objects as groups of features, more refined geometric models can be used to actually recover the objects' structure. This splits the unsupervised learning process in two steps: the first is for grouping parts that belong to the same object and the second for processing each object individually in order to recover its actual spatial structure. In this paper we focus on the first step, in the case of objects from low resolution images.

In data mining, Kubica et al. ([3], [4]) use co-occurrences for discovering groups of people in an unsupervised manner. They use generative models more appropriate for human behavior, but the concept of linking temporally dependent entities by observing their co-occurrence is similar to our approach to learning about physical objects.

1.1. Approach

Physical objects can be represented as collections of observable features that co-exist over time and also display a correlated behavior. These properties stand in high contrast with the independent behavior of features that belong to different entities.

We need to simplify the problem in order to be able to handle it practically. As in previous approaches, we represent objects as clusters of features/parts. Currently this is one of the main approaches in object representation ([1], [6]). We can use any feature of the object's appearance that can be robustly matched over time, under different in-plane rotations, changes in scale or slight changes in pose.For the purpose of this paper we are representing every feature (or keypoint) as a 128 dimensional SIFT descriptor, its location and scale, and the orientation of the main intensity gradient within its scaled neighborhood ([6]). We locate the features with the DOG detector, as developed by Lowe ([6]). There are other choices of features descriptors and detectors that could also be used ([2]).

A priori we do not know the number of objects, their range of pose, or the number of parts belonging to each object. Therefore we first focus on modelling the interdependence between pairs of parts and later use these pairwise connections to recover the whole objects.

We assume that any keypoint detected in an image is the appearance of some *part* in the real world. Then, any two parts detected in a given frame either belong to the same aspect/face of an object (and implicitly belong to the same object) or belong to different objects. There is no explicit concept of background, clutter or foreground. They are all objects and it is left to the learner to figure out how to group the parts into objects.

The algorithm is divided in three phases as follows: In the first phase (Section 2) we extract interest points and match them across frames. We process the frames sequentially, in a greedy fashion, to reduce the computational complexity. At every frame we try to match the keypoints in the current frame with clusters of matched keypoints from previous frames. We will refer to such a cluster as a *part* since it contains key points that should represent the appearances of the same physical part. At this stage we also count co-occurrences of pairs of parts within local neighborhoods, and use affine pairwise geometric constraints for invalidating possible wrong matches.

In the second phase (Section 3) we classify pairs of parts as dependent (belonging to the same aspect of an object) or independent, by using probabilistic models for the two classes. We label each pair with the MAP estimate based on that pair's co-occurrences accumulated over the sequence.

In the third phase (Section 4) we form objects as connected components of dependent parts, using the pairwise labelling done in Section 3. Then we check the robustness of the connected components obtained by using random graph sampling, and prune them accordingly if necessary.

2. Observing co-occurrences

A part $p_i = \{k_1^i, k_2^i, \ldots, k_{n_i}^i\}$ is a collection of key points $k_q = (d_q, x_q, y_q, s_q, \theta_q)$. Each key point from p_i has been extracted from some frame and included in p_i after matching p_i, as explained below; d_q is a 128 dimensional SIFT descriptor, (x_q, y_q) is the location in the image, s_q the corresponding scale and θ_q the angle of the main gradient computed within the scaled neighborhood at that location (see [6] for more details). The *geometry* (location, scale and orientation) of a key point is represented by two image points, given by $[(x_q - s_q \cos\theta_q, y_q - s_q \sin\theta_q), (x_q + s_q \cos\theta_q, y_q + s_q \sin\theta_q)]$. Therefore a pair of key points is represented as 4 points. (Figures 2(a) and 2(b)). These points are used in monitoring the affine geometry of pairs of parts.

At every frame we first try to detect parts from previous frames by matching the key points from the current frame with the parts (= clusters of key-points) collected from previous frames. These key-points will be added to the parts they match or start forming new clusters if they did not match any old ones.

For every current k_q we find its nearest part p_i, where the distance $d(k_q, p_i)$ between k_q and any part p_i is defined as the smallest L_2 norm between the SIFT descriptor of k_q and the SIFT descriptor of any of the key-points of p_i. Then we check the following conditions to insure that, given the current frame, one key point can match only one part and vice-versa, and that the matches are well separated from the non-matched pairs (similar conditions were required by Lowe [6]):

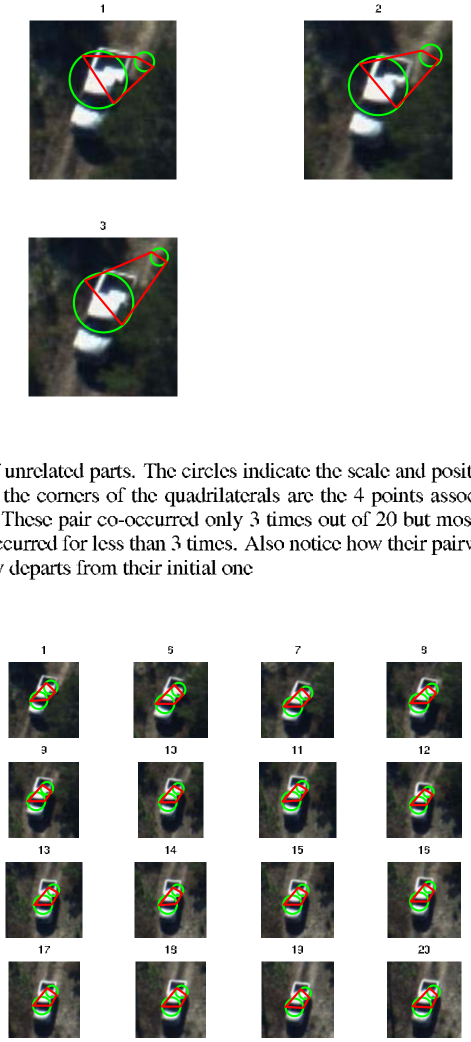
1. $d(k_q, p_i) < thresh1$. We use thresh1 = 300 (this is a loose threshold; most matches are within a distance of 150).
2. $d(k_q, p_i)/d(k_q, p_j) < thresh2$ and $d(k_q, p_i)/d(k_w, p_i) < thresh2$, where p_j is the second closest part to k_q, and k_w is the second closest key point from the current frame to p_i. We use thresh2 = 0.7.

Key points k_q that fail condition 1 are labeled as new parts only if they obey the additional constraints $d(k_q, p_i) > thresh3$, where $thresh3 > thresh1$ (we used $thresh3 = 350$). This is necessary for reducing the risk of creating new clusters from key points that have been misclassified as not belonging to old clusters.

All k_q that meet only condition 1 are discarded because they are not discriminative enough and will introduce ambiguity in later matching. This approach is similar to approaches from text classification literature where very common words like "the" and "a" are discarded. A similar approach was also taken by Sivic in [9].

The new parts and the detected parts (the ones matched by some k_q such that conditions 1 and 2 are met) are counted as co-occurring in the current image only if their affine pairwise geometry remains consistent with their previous ones, and if they remain relatively close to each other during the sequence. The details and the motivation for these constraints are discussed next.

The first assumption is that, *in general, objects occupy compact regions in space*. To take advantage of this property we will allow only parts that are relatively close to each other to establish relationships of direct dependency. Parts that are farther away from each other can still become dependent but only through dependencies with intermediate parts. The closeness of two parts is measured relative to their natural scale. The DOG detectors return a natural scale for each part. We define the neighborhood of a pair of parts (A, B) as a multiple of their average natural scale: $k * mean(s_A, s_B)$. For example, we do not want parts from one corner of a relatively large image to build dependencies with parts from the other corner unless they are indirectly dependent through other intermediate parts. In this manner the process of learning pairwise dependencies becomes invariant to the size of the input image. More precisely parts (A, B) can be considered as co-occurring in the current frame only if $\sqrt{(x_q - x_w)^2 + (y_q - y_w)^2} \le k{*}mean(s_A, s_B)$(we set $k = 5$, but different values worked as well).



(a) Pair of unrelated parts. The circles indicate the scale and positions of the two parts; the corners of the quadrilaterals are the 4 points associated with each pair. These pair co-occurred only 3 times out of 20 but most unrelated pairs co-occurred for less than 3 times. Also notice how their pairwise geometry slowly departs from their initial one

(b) Pairs of parts that belong to the same object tend to co-occur most of the time. These two parts were detected together in 16 frames out of 20

Figure 2: Dependent (b) vs independent (a) parts

The second assumption is that *objects are almost rigid locally*. Our input sequences consist of shots taken from distant view-points relative to the size of the objects (cars in our case). Since the object is seen at a low resolution, we can check reliably only simple geometric dependencies. The scene is far enough so that we can consider we have an affine camera. Therefore, the pairwise geometry of any two neighboring parts belonging to the same object (2 points from each part as explained previously) goes through affine transformations in the image plane as the view-point changes. Consequently, if two parts (A, B) detected in the current image happened to co-occur previously, we impose the additional requirement that their current pairwise geometry (defined by 4 points) has to be affine equivalent to the 4 points associated with their first co-occurrence. Similarity constraints would also work. Figures 2(b) and 2(a) illustrate how dependent parts keep a consistent pairwise geometry over time, as opposed to independent parts. We also use the geometric constraints for invalidating possible wrong matches: if a part (seen previously) is matched in the current image, it is kept as a valid detection only if it validly co-occurs (it passes the co-occurrence constraints) with at least one other part with which it co-occurred at least once before. We found that in practice this requirement filtered out most of the invalid detections.

3. Interpreting co-occurrences

In this section we look for appropriate probabilistic models in order to classify every pair of parts (i, j) as either dependent (belonging to the same aspect of an object) or independent, given their observed individual counts n_i, n_j and their pairwise count n_{ij}.

In the case of distant cameras the range of view points from which an object is seen can be represented as a sphere of unit radius. The range of view-points from which a particular point A on the object's surface is visible becomes a 2D region V_A on the view sphere (Figure 3). Pairs of points (A, B) from the same object, whose visibility areas V_A and V_B have significant overlap are likely to co-occur in images of that object taken from random view points. More precisely, if the set of view-points is sampled uniformly on the view sphere, the probability of seeing both A and B (randomly picking a view point v from $V_B \bigcap V_A$), given that we see one of the two (that is $v \in V_A \bigcup V_B$) is equal to the ratio of the overlapping area of $V_B \bigcap V_A$ and the area of the union region $V_A \bigcup V_B$: $p(v \in V_B \bigcap V_A | v \in V_B \bigcup V_A) = Arca(V_B \bigcap V_A)/Arca(V_A \bigcup V_B) = r_{AB}$. The ratio r_{AB} can be interpreted as a measure of closeness between two parts on the object, and happens to be equal to one minus the Tanimoto distance between V_A and V_B, which is a standard metric between two sets. In the presence of occlusion due to other objects or detection errors, we expect the co-occurrence probability to be smaller than r_{AB}, but one can show that it will not decrease significantly if the errors in detection or probability of occlusion due to other objects are relatively small.

In order to discriminate between pairs that highly co-occur and the pairs of unrelated features that rarely co-occur we define the notion of a *co-occurrence set*. A co-occurrence set S_{p_+} is a collection of parts from the same object such that for any two parts $(A, B) \in S_{p_+}$ the ratio $r_{AB} = Arca(V_B \bigcap V_A)/Arca(V_A \bigcup V_B) \geq p_+$.

In the absence of other information we assume that the view point on the view sphere is sampled uniformly over the input sequence such that $p(v \in V_B \bigcap V_A | v \in V_B \bigcup V_A) = r_{AB}$. Then, for any pair in the same co-occurrence set we approximate $p(A \bigcap B | A \bigcup B)$ with the lower bound p_+ and obtain a pairwise Bernoulli distribution for their conditional co-occurrence.

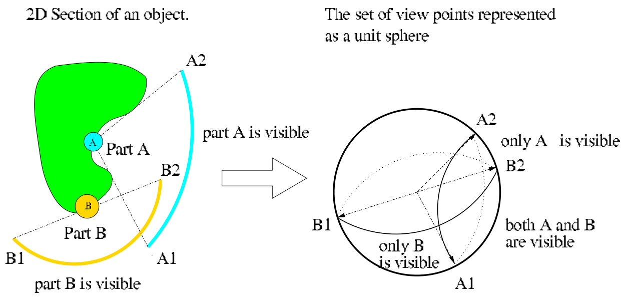


Figure 3: For a random view-point the probability that the pair (A, B) occurs when at least one of the two parts occurs is equal to the ratio of the common area of visibility and the union of their visibility areas. Small deformations of the object do not affect this ratio significantly.

In reality, the conditional $p(v \in V_B \bigcap V_A | v \in V_B \bigcup V_A)$ is different for every pair $(A, B) \in S_{p_+}$, since it depends on the parts' actual location on the object and the object's structure. In practice a lower bound approximation of the conditional probability of correlated parts is enough for discriminating them from pairs of independent parts.

We model the co-occurrence of unrelated parts $p(A \bigcap B | A \bigcup B) = p_-$ as another Bernoulli distribution. Different objects have different frequencies of co-occurrence (ex. cars from different cities vs. cars from the same neighborhood), but these frequencies are expected to be close to zero. If we think of the world of an object as a set of w disjoint sites reachable by that object, and let our object be present at a particular site with uniform probability $\frac{1}{w}$, then the expected probability of co-occurrence of two independent objects from the same world at a particular site is $p(A \bigcap B | A \bigcup B) = \frac{1}{2w-1}$. Obviously, the bigger the world considered the smaller the probability $p(A \bigcap B | A \bigcup B)$. In our experiments we fixed p_- to 0.02, which is the smallest strictly positive co-occurrence count $w_{AB} = \frac{n_{AB}}{n_A + n_B - n_{AB}}$ we can get from a 51 frame input sequence ($\frac{min(n_{AB},1)}{max(n_A + n_B - min(n_{AB},1))} = \frac{1}{N-1}$), where N is the number of frames in the sequence.

Parameter p_+ has direct influence on the size of the co-occurrence sets. The larger p_+ the smaller the co-occurrence sets. On the one hand, we want large co-occurrence sets that overlap and cover the whole objects, but on the other hand we want a large p_+ as compared to p_- such that the two Bernoulli distributions are well separated. In the experiments section we will analyze how parameter p_+ affects the performance.

As the individual occurrences of parts n_A and n_B increase we expect the relative co-occurrence $\frac{n_{AB}}{n_A+n_B-n_{AB}}$ of related parts to approach a value close to their true ratio $Arca(V_B \bigcap V_A)/Arca(V_A \bigcup V_B)$, and the co-occurrence of unrelated parts to approach zero (or a value very close to zero).

Now we have generative models for two classes: class O for pairs that belong to the same co-occurrence set and class $\neg O$ for the rest of the pairs, which we consider as independent. Pairs of parts that belong to the same object but have very little overlap in visibility cannot be discriminated from pairs of unrelated parts in a pairwise way, but only through connections with intermediate parts.

We label the pairwise links with the class $C \in \{O, \neg O\}$ that maximizes the MAP estimate.

$$C_{AB} = argmax \frac{p(n_{AB}, n_A, n_B/C)P_C}{\sum_{Q \in \{O, \neg O\}} p(n_{AB}, n_A, n_B/Q)P_Q} \tag{1}$$

We assume that the priors $P_O = P_{\neg O}$, so the MAP classification is equivalent to maximizing the likelihood $p(n_{AB}, n_A, n_B/C) = p^{n_{AB}}(1-p)^{n_A+n_B-2n_{AB}}$, where $p = p_+$ if $C = O$ and $p = p_-$ otherwise. Thus, we classify as O when $p_-^{w_{AB}}(1-p_-)^{(1-w_{AB})} < p_+^{w_{AB}}(1-p_+)^{(1-w_{AB})}$, where $w_{AB} = \frac{n_{AB}}{n_A+n_B-n_{AB}}$. Then we have:

$$\begin{aligned} & w_{AB} > \frac{\log \frac{1-p_+}{1-p_-}}{\log \frac{1-p_+}{1-p_-} + \log \frac{p_-}{p_+}} \\ & 0 < p_- < p_+ < 1 \end{aligned} \tag{2}$$

A classification threshold th is set to the right hand side of inequality 2, such that a pair (A, B) is positively labeled whenever $w_{AB} > th$. As discussed previously, the expected value of w_{AB}, for randomly sampled view-points, is equal to $Arca(V_B \bigcap V_A)/Arca(V_A \bigcup V_B)$, if A and B belong to the same object. This ratio is an invariant property of each pair of parts of the same object. In the absence of any information about the relative camera viewpoint, we use only w_{AB} to separate such pairs from pairs of independent parts. One can show that $p_- < th < p_+$, so long as $p_- < p_+$. To reliably classify the pair (A, B) we need a large enough $N_{AB} = n_A + n_B - n_{AB}$. This intuition relates to concepts from hypothesis testing. We have two simple hypotheses corresponding to the two Bernoulli distributions discussed previously. If we let the null hypothesis H_0 be that the pair (A, B) is unrelated, we want to reduce the type I error of accepting an unrelated pair (A, B) as part of the same co-occurrence set. By Chebyshev's inequality, the type I error is proportional to the variance of w_{AB} (when A and B are unrelated), which in turn is inversely proportional to N_{AB}. To reduce this error we classify (A, B) as unrelated if $N_{AB} < 10$, regardless of the value of w_{AB}.

4. Grouping parts into objects

If both pairs (A, B) and (B, C) are classified as O (same aspect/co-occurrence set) we cannot say anything about whether pair A, C belongs or not to the same aspect (or co-occurrence set) in the absence of any other information. However, what we can tell is that pair (A, C) belongs to the same object. Therefore each pair is classified independently of the others and then we can use the pairwise classification to recover whole objects as connected components of parts. If a connected component is large enough we can consider it as a set of parts forming an object. Of course, errors in the pairwise classification will affect the formation of components. However, the more connected a particular component is, the less vulnerable it is to the accidental removal or addition of edges. Therefore, for any two parts (i, j) we want to measure the confidence that they are indeed in the same component.

A connected component V can be interpreted as Bernoulli random graph with vertices V and edges $E = \{c_{ij} | p_{ij} = p(O/n_i, n_j, n_{ij}) > 0.5\}$ that defines a distribution over the undirected graphs with vertices V and edges $c_{ij} \in E$ being sampled with probability p_{ij}, independent of other edges. The posterior probabilities $p_{ij} = p(O/n_i, n_j, n_{ij})$ are estimated from the likelihoods (Section 3). Let $C(A, B)$ be the set of all such graphs in which parts (A, B) are connected. Then, the probability $p(A, B)$ that pair (A, B) is connected is equal to the probability of sampling a graph $G \in C(A, B)$ in which the pair (A, B) is connected through at least one path: $p(A, B) = \sum_{G \in C(A,B)} \prod_{c_{ij} \in E_G} p_{ij} \prod_{c_{ij} \in E - E_G} (1 - p_{ij})$. The computation time of this formula is exponential in the number of edges, which makes it impractical to implement. We decided to estimate it by simulation, which will estimate the connectivity probabilities between all pairs of vertices in the connected component with good accuracy in $O(S * |E|^{3/2})$ steps, where S is the number of samples. We generate random graphs G_k , $k = 1 \ldots S$ by forming the edge matrix E_k such that $E_k(i, j) = 1$ with probability p_{ij} and 0 otherwise. Next we form the connectivity matrix $C_k = sgn((E_k)^d)$. Then $c_{ij} = 1$ if the pair (i, j) is connected within at most d edges and 0 otherwise (we choose $d = 2^3$, so C_k can be computed with three matrix multiplications). We form the matrix $P_S = \frac{\sum_k C_k}{S}$. The value $P_S(A, B)$ will be a lower bound estimate of the probability $p(A, B)$. The standard deviation of this estimate is bounded above by: $std(P_S(A, B)) = \frac{1}{S^{1/2}}(p(A, B)(1 - p(A, B)))^{\frac{1}{2}} \leq \frac{1}{2 * S^{1/2}}$. We choose $S = 1000$.

Most connected components obtained from our input sequences were strongly connected, as indicated by high values (over 70%) of $P_S(A, B)$ for all pairs in the same component. This is explained by the fact that the vast majority of pairs from the same component (95%) were connected within a maximum of 3 edges (Figure 4).

There are other types of sequences for which larger components can be formed. Imagine filming a large nearby building as we slowly move around it. In this case parts from one side of the building will be connected to parts from the other side through a large number of edges, forming a large connected component. In this case the values of $P_S(A, B)$ will discriminate between distant parts that are connected through a large number of paths (high value of $P_S(A, B)$) and distant parts connected through only one or a few paths. Since a lower value of $P_S(A, B)$ indicates a higher vulnerability to accidental errors on edges, we can make the components more robust by keeping only the maximal subset in which every two parts have $P_S(A, B) > pThresh$.

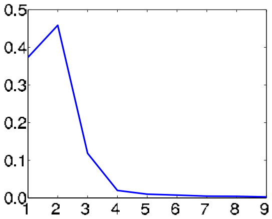


Figure 4: distribution of the distance (in number of hops) between pairs of connected parts from 25 input sequences

5. Experimental Analysis

In this section we look at the overall performance of the algorithm, while examining quantitatively the variations in performance as we vary the parameters affecting the unsupervised learning module (p_- and p_+) and the size of the input images relative to the objects size.

For each experiment we were interested whether a particular object was learned correctly from a particular sequence. We manually assigned ground truth labels to all the parts from the input sequence (clusters of key-points collected and matched by the algorithm), thus forming the ground truth sets G_1 (object parts) and G_0 (the rest of the parts). Then we picked the connected component C_1 formed by the algorithm, which had the largest overlap with the ground truth set G_1 (the true object parts). All those parts that belonged to the intersection of C_1 and G_1 were considered true positives (correctly classified as belonging to the object of interest). The parts outside the union of C_1 and G_1 were the true negatives, while the parts in their set differences were the misclassified ones ($G_1 - C_1$ the false negatives and $C_1 - G_1$ the false positives).

We wanted p_- to be close to the smallest strictly positive w_{AB} from a 30 to 60 frames sequence, so we chose 0.02 (Section 3). While it is clear that p_- should take a small value, p_+, which controls the relative overlap between the visibility areas of dependent parts on the viewing sphere, could in theory take any value between 0 and 1. We expect that the larger p_+ the smaller the number of false positives (unrelated parts will be hardly grouped together), but the higher the number of false negatives (= the number of positive parts left outside of the object's component). Therefore, we examined more closely how the performance varies with p_+. We varied $p_+ \in 0.07 \ldots 0.97$ for all sequences, then for every sequence we generated the ROC and classification error curves, as functions of p_+.

We tested the algorithm on 25 different sequences. They varied in length (17 to 61 frames), frame rate (1 to 6 frames per sec) , absolute temporal length (3 to 50 sec), absolute image sizes (120 x 120 to 500 x 400) and relative image to object size (4 to 70x the object size). The objects in the sequences were turning, passing by each other, and occluding one another partially or sometimes totally (Figure 5). The processing time per frame ranged from 0.3 to 2 sec in Matlab, on a 2Ghz Pentium processor.

Figure 5 shows the results obtained on four different sequences all sampled at 4 frames per sec. Each sequence is of a different type. Sequence 1 is particularly difficult due to the relatively large amount of clutter (frame size 500 x 370, object size approximately 35 x 60, 500-600 key points per frame, about 10 key points on the car). The clutter is not well separated spatially from the object (the car is driving through a forest road, with rich ground texture and vegetation nearby), but the temporal co-occurrences within local neighborhoods separated the car successfully from its surroundings. Sequence 2 contains a truck that turns 180 degrees. The algorithm is able to learn different aspects of the truck and connect them together, while separating them from another distracting car that is present in the sequence for more than half the number of frames. In Sequence 3 five vehicles that pass by each other are successfully learned simultaneously, without specifying a priori how many objects were present in the sequence. In Sequence 4 a cabriolet undergoes different degrees of occlusion and it is extracted correctly.

Figure 6(a) shows the mean performance curve (solid line) and the curve at one standard deviation below the mean (dashed line), computed from tests on all 25 sequences. The high mean and small standard deviation indicates that the algorithm performs robustly in a variety of situations.

An important aspect of the performance evaluation was to examine how the classification error varies with p_+ (Figures 6(b) and 6(a)). On most sequences the performance increases suddenly as p_+ approaches 0.3. For values larger than 0.3 the classification error stabilizes at around $6 - 7\%$. The error curve and small standard deviation prove that the performance is not sensitive to the parameter p_+ for values larger than 0.3. This result is supported by the distribution of co-occurrence weights w_{AB} on pairwise links from the data (all 25 sequences). Using the ground truth labeling of individual parts, we collected histograms of the weights of pairs that belong to the same object and of pairs that belong to different objects. The two normalized histograms obtained are displayed in Figure 6(c). Around the value of 0.11 the histogram of independent weights (dashed line)

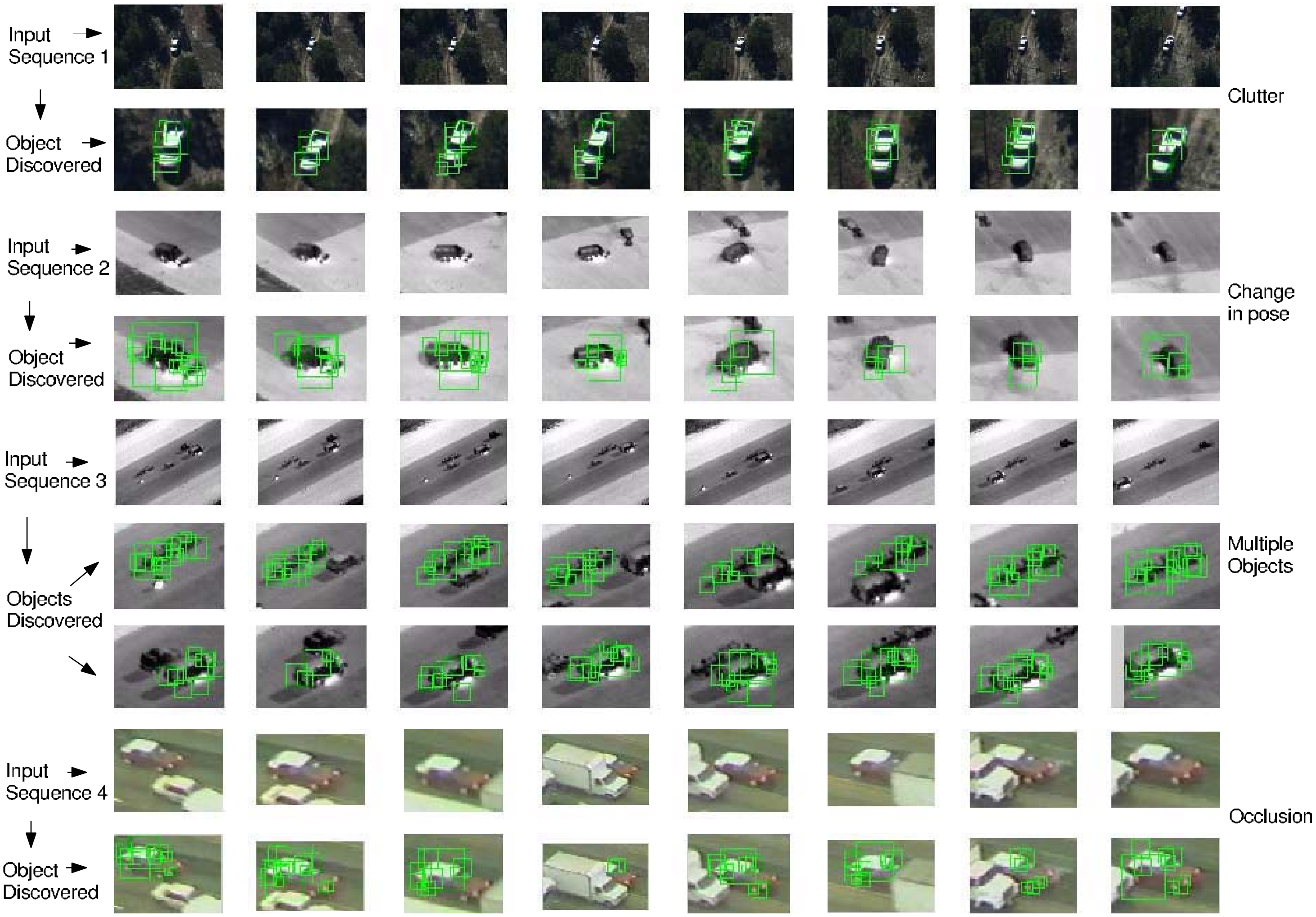


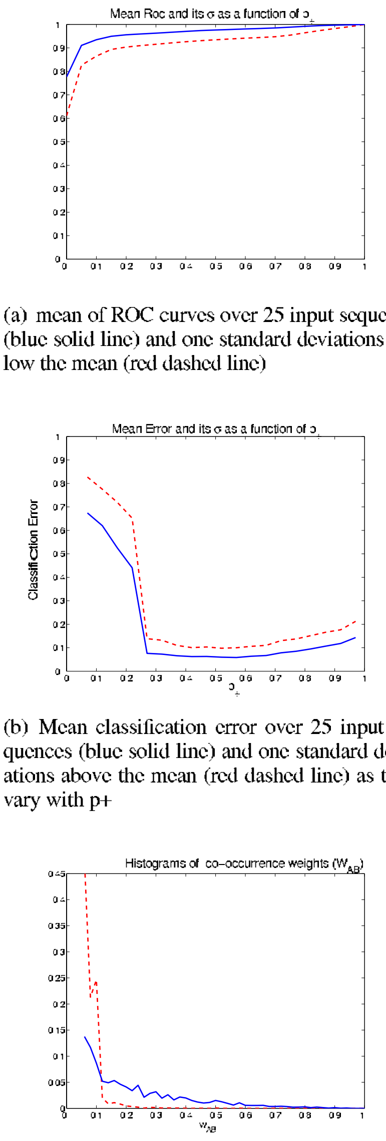
Figure 5: Results on different video sequences

drops almost vertically to near zero, below the histogram of weights of dependent pairs (solid line), which continues to decrease slowly. This suggests that a major increase in performance will occur as the threshold on weights increases above 0.11. That threshold (0.11) corresponds to $p_+ = 0.3$ (for $p_- = 0.02$), which validates the sudden jump in performance at $p_+ = 0.3$ obtained in experiments (Figure 6(b)).

The third aspect of the evaluation was to monitor how the size of the image relative to object size affects the performance. We ran this experiment on three different videos. (Figure 5, sequences 1,2 and 3. Sequence 2 shows the smallest size window, the others show the largest size). For each video we selected 3 input sequences from the same frames in the video (at 5 frames per second), but different window sizes. The sizes varied from 12x to about 60-70x the object size. The results (Figure 7) show that the performance does not deteriorate as the window size increases. We believe that the unsupervised learning module is not sensitive to the amount of clutter or the number of objects to be learned from the scene, so long as the objects behave independently. The overall performance will be affected, however, due to the fact that the errors in the matching of interest points increase considerably as the window size increases.

6. Conclusions

We have presented an algorithm that discovers objects in an unsupervised manner from video sequences, by modelling explicitly the dependent temporal co-occurrences of parts belonging to the same object vs the independent behavior of unrelated parts. The algorithm performs successfully in a variety of complex situations, so long as the different objects behave independently and the matching of parts over



(a) mean of ROC curves over 25 input sequence (blue solid line) and one standard deviations below the mean (red dashed line)

(b) Mean classification error over 25 input sequences (blue solid line) and one standard deviations above the mean (red dashed line) as they vary with p+

(c) Normalized histograms of co-occurrence weights for pairs that belong to the same object (blue solid line) and pairs that belong to different objects (red dashed line) over all 25 sequences

Figure 6: Performance evaluation

frames is reliable. As future work, we have started to explore how the relationships learned between parts could be used to refine the matching process, and to integrate the matching and learning modules into the same probabilistic framework.

7. Acknowledgments

We thank Martial Hebert for helpful comments and insightful discussions. This work was sponsored by DARPA as part of the VIVID project through Grant NBCH1030013. The content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

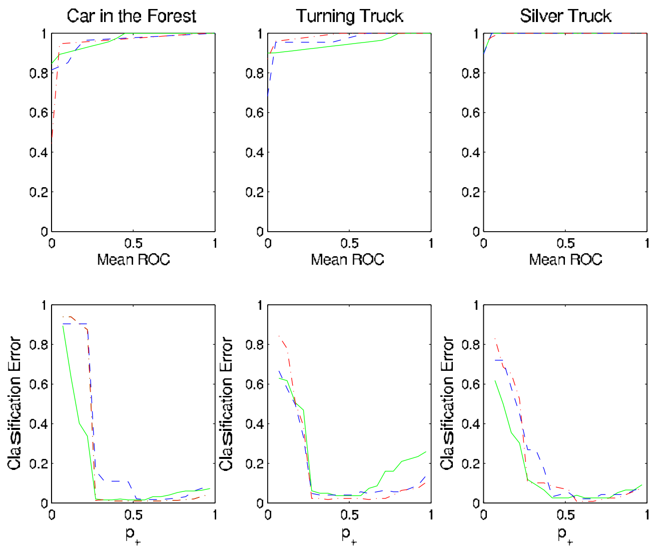


Figure 7: Mean ROC and Error curves for three different video sequences as we vary the size of the input images (12x (green solid), 30x (blue dashdot), 60x (red dashed) the object size)

References

[1] R.Fergus, P.Perona, A. Zisserman "Object Class Recognition by Unsupervised Scale-Invariant Learning." In Proc. CVPR, pp. II: 264–271, 2003.

[2] T. Kadir, M. Brady "Saliency, Scale and Image Description", In Proc. ICCV, pp. II: 83-105, 2001

[3] J. Kubica, A. Moore , J. Schneider "Finding Underlying Connections: A Fast Graph-Based Method for Link Analysis and Collaboration Queries", In Proc. ICML, pp. 392-399, 2003

[4] J. Kubica, A. Moore, Jeff Schneider "K-groups: Tractable Group Detection on Large Link Data Sets" In Proc. ICDM, pp. 573-576, 2003

[5] S. Lazebnik, C. Schmid, J. Ponce "Affine-Invariant Local Descriptors and Neighborhood Statistics for Texture Recognition" In Proc. CVPR, pp. 649-655, 2003

[6] David G. Lowe " Object Recognition from Local Scale-Invariant Features " In Proc. ICCV, pp 1150-1157, 1999

[7] D. Ramanan, D. A. Forsyth "Using Temporal Coherence to Build Models of Animals", In Proc. ICCV, pp. I: 338, 2003

[8] Fred Rothganger, S. Lazebnik, C. Schmid, J. Ponce "Segmenting, Modeling, and Matching Video Clips Containing Multiple Moving Objects" In Proc. CVPR, pp II: 914-921, 2004

[9] J. Sivic, A. Zisserman "Video Data Mining Using Configurations of Viewpoint Invariant Regions", In Proc. CVPR pp. 488-495, 2004