Conditional Random Fields for Multi-Camera Object Detection

Gemma Roig*[1,2] Xavier Boix*[1,2]
[1]ETHZ, Zurich (Switzerland)
{gemmar,boxavier}@vision.ee.ethz.ch

Horesh Ben Shitrit[2] Pascal Fua[2]
[2]EPFL, Lausanne (Switzerland)
{horesh.benshitrit,pascal.fua}@epfl.ch

Abstract

We formulate a model for multi-class object detection in a multi-camera environment. From our knowledge, this is the first time that this problem is addressed taken into account different object classes simultaneously. Given several images of the scene taken from different angles, our system estimates the ground plane location of the objects from the output of several object detectors applied at each viewpoint. We cast the problem as an energy minimization modeled with a Conditional Random Field (CRF). Instead of predicting the presence of an object at each image location independently, we simultaneously predict the labeling of the entire scene. Our CRF is able to take into account occlusions between objects and contextual constraints among them. We propose an effective iterative strategy that renders tractable the underlying optimization problem, and learn the parameters of the model with the max-margin paradigm. We evaluate the performance of our model on several challenging multi-camera pedestrian detection datasets namely PETS 2009 [5] and EPFL terrace sequence [9]. We also introduce a new dataset in which multiple classes of objects appear simultaneously in the scene. It is here where we show that our method effectively handles occlusions in the multi-class case.

1. Introduction

Many state-of-the-art object detection schemes [21, 3, 7] rely on sliding a window across the image and, for each location, running a classifier to decide whether or not an object of interest is present within it. This usually yields multiple responses around every single true positive and a post-processing, often in the form of non-maxima suppression, is often required. Furthermore, these schemes rarely work well in the presence of occlusions, which are frequent in crowded scenes.

One way around this problem is to connect detections across time and to use temporal consistency for disambiguation purposes [15, 17, 16, 18, 22, 1]. Another is to use multiple synchronized cameras, which is useful when images can only be acquired at a too low rate for temporal consistency to be sufficient, and when the scene becomes so crowded that single-camera solution are no longer effective [19].

In this paper, we investigate the latter approach, which has received surprisingly little attention in the Computer Vision literature. Given binary background/foreground masks computed in views from multiple registered cameras, [9] showed that a generative model could be used to compute probabilities of presence of people at various locations while handling occlusions in a principled way. That approach, however, could only work in cases where a background subtraction algorithm could be expected to work well, which limits its applicability. It was later extended to replace the background subtraction algorithm by the output of a classifier [2], but this required modelling the responses of the classifier depending on the set up of the scene and the occlusions, and is usually computationally unaffordable.

We introduce a framework based on Conditional Random Fields (CRF) [14]. Given the classification score returned by the object detectors in each location, such as those depicted by Figure 1, we introduce a CRF to evaluate the labeling of the discretized ground plane indicating the presence of the objects. To handle an occlusion map for each view, we define in our CRF a different set of nodes per view, and a potential that models the occlusions. Using a consistency potential among views, the CRF encourages reaching an agreement of the occupancy and occlusion maps. Our model simultaneously predicts the presence of objects in all locations of the scene in a principled manner. We introduce an iterative algorithm that makes the underlying optimization problem feasible, and we learn the model using the max-margin paradigm.

We test our approach on datasets available on-line, namely the PETS09 [5] and EPFL terrace sequence [9]. Since these multi-view benchmark datasets do not deal with multiple object-classes, and are amenable to background subtraction because they do not incorporate strong lighting changes, we also created a more challenging dataset to test our method when this is not the case. We will show that we

[*] Both authors contributed equally. Work mainly done at EPFL.

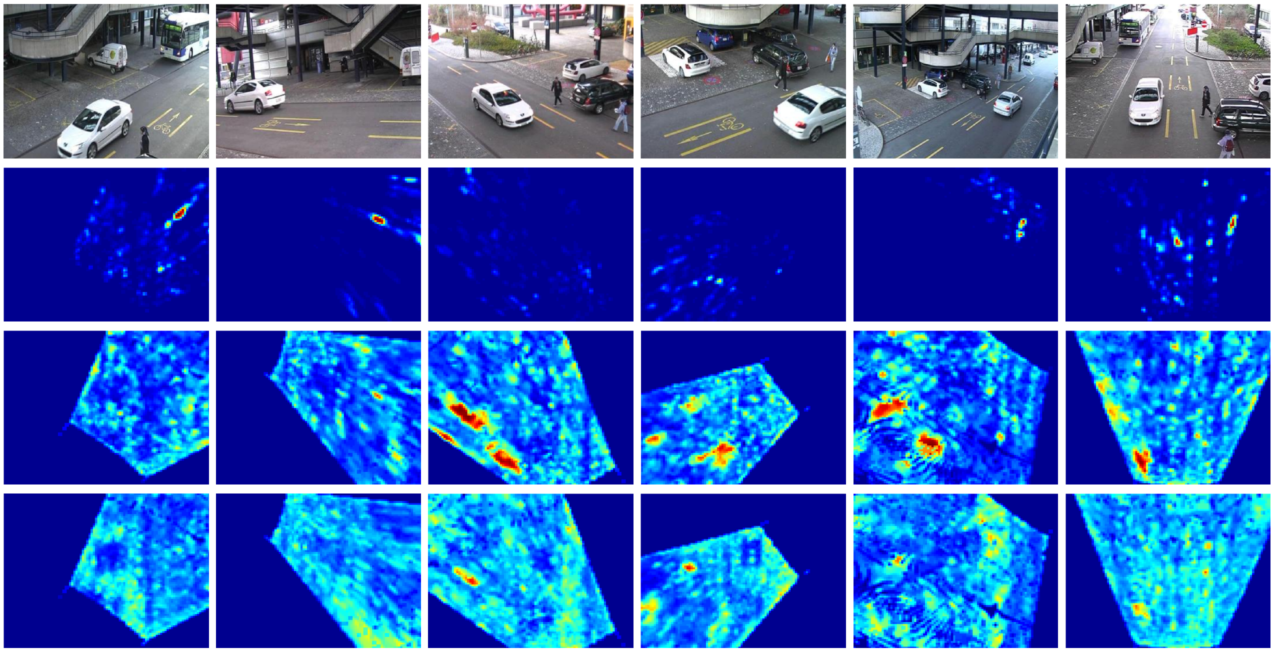


Figure 1. Example of the multi-camera setup from the dataset introduced in this paper. In the first row, we show the images acquired with the 6 different cameras. We also show the responses of the classifiers for the different object classes (pedestrians, cars and buses). This is the input of our algorithm. We estimate a labeling for each view in order to handle different view-dependent occlusion maps. The final result correspond to the labeling at the top view which has been obtained enforcing consistency among views.

achieve good results in all of them.

2. Model

In this section we introduce our formulation for the multi-camera object detection. This formulation aims at estimating the presence of objects given the responses of windows-based object detectors computed on several viewpoints of the scene. We consider that all target objects lie on the ground plane, and the calibration of all cameras is known. The ground plane has been discretized in a predefined set of cells. Each cell location has a single bounding box associated for each view and object class. For instance, in Figure 2 we indicate in red the bounding boxes associated to the cell location marked with a big white dot on the ground plane. We select such bounding boxes by taking the ones with maximum detection score from the set of possible bounding boxes of the cell (in blue). We do so for each cell in the ground plane. Thus, we predefine the bounding boxes by selecting them with the detection score, which is in contrast to previous methods that impose hard constraints on the shape of the object, *e.g.* [9] considers that persons occupy bounding boxes of fixed size.

Our goal is to indicate the center of the objects in the ground plane given the detection scores such as in Figure 1. It is known that a main point to effectively perform the multi-camera detection task is the modeling of the occlusions between objects. To model such occlusions, we label the ground plane indicating that there is either the center of an object or not, and also the occluded locations. We model the probability density function of how likely is a certain labeling of the ground plane with a Conditional Random Field (CRF) [14]. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the graph that represents our CRF, where $\mathcal{V}$ is used for indexing the nodes that correspond to random variables, and $\mathcal{E}$ is the set of undirected edges representing compatibility relationships between random variables. We use $\mathbf{X}$ to denote the set of random variables, and $\mathbf{x}$ a possible state or instantiation of $\mathbf{X}$.

A clique is a subgraph in which every node is connected to all other nodes in the subgraph. Let $\mathcal{C}$ be the set of cliques of the CRF that are not a subset on any other clique, which are also known as maximal cliques. Then, the energy function of the CRF is $E(\mathbf{x}) = \sum_{c \in \mathcal{C}} \varphi_c(\mathbf{x}_c, \mathbf{O})$, where φ_c is the potential function of the maximal clique $c \in \mathcal{C}$, and $\mathbf{O}$ some observations or measurements. From now on, we omit the dependency of the potentials on $\mathbf{O}$ for notation simplicity. Finally, let $\mathbf{x}^*$ be the state that minimizes the energy function, $\mathbf{x}^* = \arg\min_{\mathbf{x}} E(\mathbf{x})$.

Each random variable takes a discrete value from a set of labels $\mathcal{L}$. We have a label for each kind of object ($\mathcal{L}_{\text{object}} = \{l_1, \ldots, l_m\}$), and also a label to indicate that the location is empty ($l_\varnothing$) or that it is occluded (l_{occ}). Recall that the occlusions generated between objects are dependent on the viewpoint, and therefore, there is a different occlusion map for each view. In our approach, we use a different set of

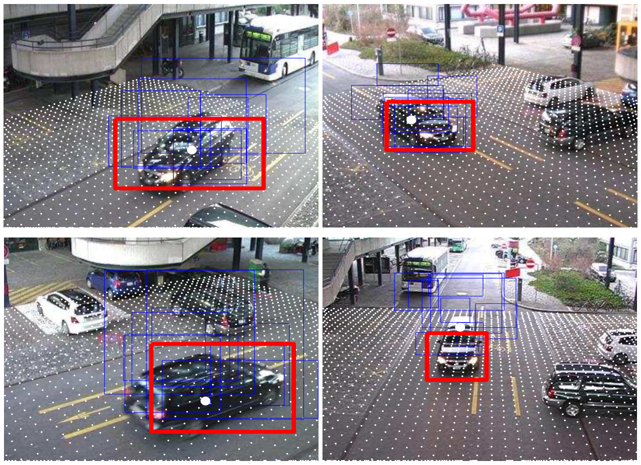


Figure 2. We represent objects with its location on the ground plane (denoted with a white dot) and a bounding box for each view (in red). For each location there are several candidate bounding boxes (denoted in blue). We select the bounding box with highest score in each view.

random variables for each view which enables us labeling the occlusions depending on the viewpoint. Each random variable might be indexed using two indices, one for the set of views $\mathcal{P}$ and another for the set of ground plane locations $\mathcal{Q}$, which gives rise to $\mathcal{V} = \mathcal{P} \times \mathcal{Q}$. For a view $v \in \mathcal{P}$, we denote as x_i^v a random variable associated with the location indexed with $i \in \mathcal{Q}$. Analogously, we define $\mathcal{N}_i^v$ as the set of neighbors of random variable x_i^v which are in the same view $v \in \mathcal{P}$.

In our approach, we consider all N views with a camera associated and also the top view, yielding a set of views as $\mathcal{P} = \{v_{top}, v_1, v_2, \ldots, v_N\}$. Although we never have a camera in the top view, our main goal can be translated as estimating the most probable labeling of the ground plane seen from the top view. Thus, we jointly infer the most likely labeling of all random variables at all different views, but the output of the algorithm is the labeling at the top view.

The energy function $E(\mathbf{x})$ is defined as the sum of the unary, neighboring, occlusion and consistency potentials:

$$
\begin{aligned}
E(\mathbf{x}) = \\
\underbrace{\sum_{v\in\mathcal{P}, i\in\mathcal{Q}} \phi_i^v(x_i^v)}_{\text{unary}} + \underbrace{w_n \sum_{v\in\mathcal{P}, i\in\mathcal{Q}, j\in\mathcal{N}_i^v} \psi(x_i^v, x_j^v)}_{\text{neighboring}} + \\
\underbrace{w_o \sum_{v\in\mathcal{P}, i\in\mathcal{Q}\setminus v_{top}} \Lambda(x_i^v, \{x_j^v\}_{j\in\mathcal{K}_i^v})}_{\text{occlusion}} + \underbrace{w_c \sum_{i\in\mathcal{Q}} \vartheta(\{x_i^v\}_{v\in\mathcal{P}})}_{\text{consistency}},
\end{aligned}
\tag{1}
$$

where $\mathcal{K}_i^v$ are all locations in the view that can occlude the location $i \in \mathcal{Q}$, and w_n, w_o, w_c weight the potentials. In Figure 3 we show the graph representation of our model, indicating the notation used in this paper. The unary term ϕ_i^v encodes the detection scores for each location and view. The neighboring term ψ determines the pairwise relationship between neighboring nodes in $\mathcal{N}_i^v$. It represents a penalization for the labeling of two connected nodes and it is able to enforce that the center of certain objects are usually not close to each other. The occlusion term Λ is a high-order potential that determines the cost of labeling a node as occluded. Finally, the consistency potential ϑ expresses the dependency relationship between labels of the nodes that represent the same ground plane location in different views: It enforces an agreement of the labeling among views. All parameters in the energy function are learned following the max-margin paradigm as explained in Section 4. In the following we explain in detail the potentials of the energy function $E(\mathbf{x})$.

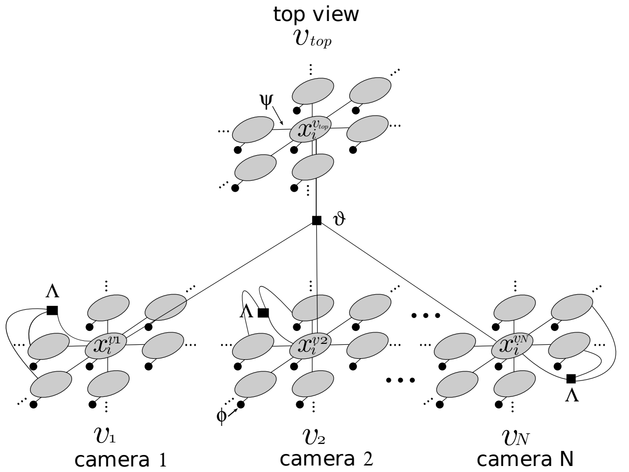


Figure 3. Graph representation of our model. For each location $i \in \mathcal{Q}$ we define a random variable x_i^v per camera $v_1, v_2, \ldots, v_N \in \mathcal{P}$ plus a random variable for the top view $v_{top} \in \mathcal{P}$. Each random variable has a unary potential ϕ associated. ψ is the neighboring potential for each view. Λ is a high order term that evaluates labeling as occlusion x_i^v taking into account the locations $\mathcal{K}_i^v$. ϑ is the consistency potential among views, which encourages an agreement between them.

The unary term ϕ_i^v is based on the scores obtained from the object detector independently computed at every location and view, referred as $\mathbf{s}_{x_i^v}$. Since in the top view we do not have a camera, it becomes

$$
\phi_i^v(x_i^v) = \begin{cases} 0 & \text{if } v = v_{top} \\ \mathbf{w}_{x_i^v}^T \mathbf{s}_{x_i^v} & \text{otherwise} \end{cases}, \tag{2}
$$

where $\mathbf{w}_{x_i^v}$ are the weighting parameters for class x_i^v. The scores for empty and occlusion classes are set to 0. To learn biases between different object classes, we append a constant 1 to make $\mathbf{s}_{x_i^v}$ two-dimensional.

The neighboring potential ψ evaluates compatibility of labeling x_i^v and x_j^v to the random variables X_i^v and X_j^v which are neighbors in the ground plane. Its purpose is to introduce the avoidance strategy between object centers. The neighboring $\mathcal{N}_i^v$ of location i is defined taking the locations that are closer than a certain distance. We penalize when there is more than an object in the neighboring, *i.e.* for a $j \in \mathcal{N}_i^v$:

$$\psi(x_i^v, x_j^v) = \mathrm{T}[x_i^v, x_j^v \in \mathcal{L}_{\text{object}}], \tag{3}$$

where $\mathrm{T}[\cdot]$ is the indicator function, and $\mathcal{L}_{\text{object}}$ the set of object labels (*i.e.* $\mathcal{L} \setminus \{l_{\varnothing}, l_{\text{occ}}\}$).

The occlusion term $\boldsymbol{\Lambda}$ is a high-order potential that determines the cost of labeling as occluded a certain random variable, excluding the ones associated to the top view. Recall that $\mathcal{K}_i^v$ are all locations in view $v \in \mathcal{P}$ that can occlude the location $i \in \mathcal{Q}$. We consider that this is the case when the occluding bounding box is in front and overlaps more than a certain threshold. This threshold depends on the performance of the object detector on detecting partially occluded objects.

The occlusion potential encourages labeling a location as occluded when there is a labeled object in at least one location in $\mathcal{K}_i^v$. Thus, the occlusion term becomes

$$\Lambda(x_i^v, \{x_j^v\}_{j \in \mathcal{K}_i^v}) = \mathrm{T}[x_i^v = l_{\text{occ}}] \oplus \max_{j \in \mathcal{K}_i^v} \left\{ \mathrm{T}[x_j^v \in \mathcal{L}_{\text{object}}] \right\}, \tag{4}$$

where $\oplus$ the exclusive or operator (XOR). Analyzing Eq. (4) we see that the maximum taken over all locations in $\mathcal{K}_i^v$ returns 1 when there is at least one object occluding the cell, and the XOR returns 1 when the cell is not labeled accordingly. Note that it is a high-order potential because it jointly takes into account the labeling of all locations in the view that potentially can produce an occlusion. In the sequel we introduce an approximation of this high-order potential that enables us to effectively infer a labeling of the ground plane.

The consistency potential ϑ enforces coherence between views. Roughly, it encourages assigning the same labeling in the top view to all other views unless there is an occlusion. It is composed by the sum of two terms, a pairwise and a high-order potential respectively:

$$\vartheta(\{x_i^v\}_{v \in \mathcal{P}}) = \sum_{v \in \mathcal{P} \setminus v_{top}} \vartheta^P(x_i^v, x_i^{v_{top}}) + \vartheta^H(\{x_i^v\}_{v \in \mathcal{P}}). \tag{5}$$

The first term, ϑ^P, is the sum of pairwise potentials between a random variable in the top view $x_i^{v_{top}}$ and the corresponding random variable in another view x_i^v. It becomes

$$\vartheta^P(x_i^v, x_i^{v_{top}}) = \mathrm{T}[x_i^v \neq x_i^{v_{top}}]\mathrm{T}[x_i^v \neq l_{\text{occ}}]. \tag{6}$$

We use $\mathrm{T}[x_i^v \neq x_i^{v_{top}}]$ to penalize assigning different labels to the random variables, and hence, we encourage a consensus among views through the top view. $\mathrm{T}[x_i^v \neq l_{\text{occ}}]$ returns 0 in case of occlusion and cancels any possible penalization. This is because when x_i^v is labeled as occluded it shall not be taken into account for the consensus. The second term, ϑ^H, is a high-order potential that determines the cost of labelling as occluded a random variable in the top view, which we only consider possible when the same location for all views is also occluded. It becomes

$$\vartheta^H(\{x_i^v\}_{v \in \mathcal{P}}) = \mathrm{T}[x_i^{v_{top}} = l_{\text{occ}}] \oplus \min_{v \in \mathcal{P} \setminus v_{top}} \{\mathrm{T}[x_i^v = l_{\text{occ}}]\}. \tag{7}$$

The minimum taken over all views returns 1 when all random variables are labelled as occlusion, and the XOR returns 1 when the top node is not labelled accordingly.

The capacity of our model of handling occlusions stems from using high-order potentials, Λ and ϑ^H. These high-order potentials, though being necessary, yield to intractable inference in practice due to the number of random variables involved. In the next section, we introduce an iterative inference algorithm able to effectively obtain a solution.

3. Inference

Minimizing the energy function $E(\mathbf{x})$ is in general NP-hard. Algorithms like Believe Propagation (BP) [13] or tree-reweighted message passing (TRW) [12] can effectively approximate a solution in practise when the energy uses potentials that involves few variables, and the cardinality of the label set does not explode exponentially. Since our energy has high-order potentials, recall Λ and ϑ^H, typical inference methods are unable to be applied successfully.

To overcome this problem, we propose an iterative algorithm (see Algorithm 1). At each iteration it finds a partial solution $\mathbf{x}_t^\star$ by inferring from an approximated energy function, referred as $\tilde{E}(\mathbf{x}, \{C_i^v\}, \{C_i^{v_{top}}\})$. The approximated energy function has the high-order potentials set with the constants $\{C_i^v\}$, $\{C_i^{v_{top}}\}$, *i.e.*

$$\tilde{\Lambda}(x_i^v, C_i^v) = \mathrm{T}[x_i^v = l_{\text{occ}}] \oplus C_i^v \tag{8}$$

$$\tilde{\vartheta}^H(\{x_i^v\}_{v \in \mathcal{P}}, C_i^{v_{top}}) = \mathrm{T}[x_i^{v_{top}} = l_{\text{occ}}] \oplus C_i^{v_{top}}, \tag{9}$$

which reduces the high-order terms to unary, and hence, enables us to minimize the approximated energy function using, for instance, BP. The constants $\{C_i^v\}$, $\{C_i^{v_{top}}\}$ are updated at each iteration using the previous partial solution $\mathbf{x}_{t-1}^\star$, and become

$$C_i^v = \max_{j \in \mathcal{K}_i^v} \left\{ \mathrm{T}[x_j^{\star v} \in \mathcal{L}_{\text{object}}] \right\} \tag{10}$$

```
Initialize($\mathbf{x}_t^\star$)
repeat
    $\mathbf{x}_{t-1}^\star = \mathbf{x}_t^\star$
    $C_i^v = \max_{j \in \mathcal{K}_i^v} \left\{ \mathrm{T}[x_j^v \in \mathcal{L}_{\text{object}}] \right\}, \forall i \in \mathcal{Q}, v \in \mathcal{P} \setminus v_{top}$
    $C_i^{v_{top}} = \min_{v \in \mathcal{P} \setminus v_{top}} \left\{ \mathrm{T}[x_i^v = l_{\text{occ}}] \right\}, \forall i \in \mathcal{Q}$
    $\mathbf{x}_t^\star$=arg min$_{\mathbf{x}}$ $\tilde{E}(\mathbf{x}, \{C_i^v\}, \{C_i^{v_{top}}\})$
until $E(\mathbf{x}_t^\star) \geq E(\mathbf{x}_{t-1}^\star)$ ;
```

Algorithm 1: Iterative algorithm used for inference. At each iteration we compute C_i^v from the partial solution $\mathbf{x}_t^\star$, which enables us to approximate the high-order potentials and compute a new partial solution with Belief Propagation.

$$C_i^{v_{top}} = \min_{v \in \mathcal{P} \setminus v_{top}} \left\{ \mathrm{T}[x_i^{\star v} = l_{\text{occ}}] \right\}. \tag{11}$$

In this way, at each iteration we update the constants $\{C_i^v\}$, $\{C_i^{v_{top}}\}$, and then infer a new partial solution $\mathbf{x}_t^\star$. The algorithm proceeds iteratively until the energy does not decrease. In experiments section we show that our algorithm performs well in practice, and provides a good approximation of the solution.

4. Learning

We formulate the learning of the parameters of our CRF using the max-margin framework [20]. Recall that the CRF energy function is expressed in terms of all cliques $\sum_{c \in \mathcal{C}} \varphi_c(\mathbf{x}_c)$. To formulate the learning requires expressing the potentials as $\varphi_c(\mathbf{x}_c) = \mathbf{w}^T \varphi'_c(\mathbf{x}_c)$. Thus, the energy function can be written as $\mathbf{w}^T \mathbf{\Psi}(\mathbf{x})$, where $\mathbf{w}$ are the parameters to learn, and $\mathbf{\Psi}(\mathbf{x}) = \sum_{c \in \mathcal{C}} \varphi'_c(\mathbf{x}_c)$. The parameters in our CRF are the weighting of the object classification scores $\mathbf{w}_{x_i}$, and w_n, w_o, w_c of Eq. (1).

To learn the model $\mathbf{w}$, we take several labeled images as training (we add a superindex k in our notation to index the images) and we formulate the following learning problem:

$$\min_{\mathbf{w}, \xi \geq \mathbf{0}} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_n \xi_n$$
$$\textit{s.t. } \forall k, \mathbf{x} : \mathbf{w}^T \mathbf{\Psi}^k(\mathbf{x}_{gt}^k) \leq \mathbf{w}^T \mathbf{\Psi}^k(\mathbf{x}) - \Delta(\mathbf{x}_{gt}^k, \mathbf{x}) + \xi_n, \tag{12}$$

where $\mathbf{x}_{gt}^k$ is the ground truth labeling for image k and Δ the loss function that compares the ground truth with a labeling hypothesis $\mathbf{x}$. The loss function for one image is $\Delta_k = \sum_i T[x_i^{top} \neq x_{i,gt}^{top}]$, which adds 1 for each mislabeling with the ground truth in the top view. The minimization problem in Eq. (12) introduces a constraint for every possible wrong labeling and image, which is usually intractable. However, we can use the cutting plane algorithm to approximate a solution [10]. It iteratively builds a small subset of constraints, which turn the optimization problem tractable. At each iteration we find the most violated constraint for each image, and then they are included in the set of constraints. As in [4], we find the most violated constraint inferring from the energy function adding in the unary potential the cost of the loss function.

5. Experiments

We evaluate our method on two different kinds of scenarios. First, we use a multi-view detection setup with pedestrians as the only object class to detect. We use online available datasets that include images of the same scene taken from different points of view, namely PETS09 [5] and EPFL terrace sequence [9]. To evaluate our approach when detecting more than one object class, we introduce a new dataset which is also multi-view but it includes multiple object classes (pedestrians, cars and buses) that can appear in the same image.

5.1. Datasets

Multi-camera pedestrian detection

For this task, we use several public available datasets which are for multi-view people detection using several calibrated cameras: PETS09 [5] and EPFL terrace sequence [9]. In all sequences we use the ground truth provided by [9].

The *PETS09 S2/L1* dataset[1] is a sequence of 1 minute and 54 seconds at 7fps that uses 7 cameras covering an area of approximately 100m x 30m. It used four DV cameras placed at about 2 meters above the ground, and three video surveillance cameras located between 3 to 5 meters high, and significantly far from the scene. It was filmed on a corner of a road and about 10 people can appear at the same time. We evaluate it every 5 frames from a total of 795.

The *EPFL terrace sequence*[2] consist of 5010 synchronized frames acquired with 4 DV cameras covering an area of 7m x 10m. It is an outdoor scene in which up to 9 people can appear. We evaluate it every 25 frames.

Notice that in these datasets the only object class that appears are pedestrians, and most previous approaches used background subtraction for the multi-view detection [15, 17, 16, 9]. To emphasize the strengths of our method, we introduce a new dataset that includes more than one object class.

Multi-camera multi-object detection

We introduce a new benchmark for multi-camera multi-object detection. To our knowledge, this is the first benchmark that addresses this problem. Our dataset consists of 23 minutes and 57 seconds of synchronized frames taken at 25fps from 6 different calibrated DV cameras. One camera was placed about 2m high of the ground, two others where

[1] PETS09 dataset available at http://www.cvg.rdg.ac.uk/PETS2009

[2] EPFL dataset available at http://cvlab.epfl.ch/data/pom

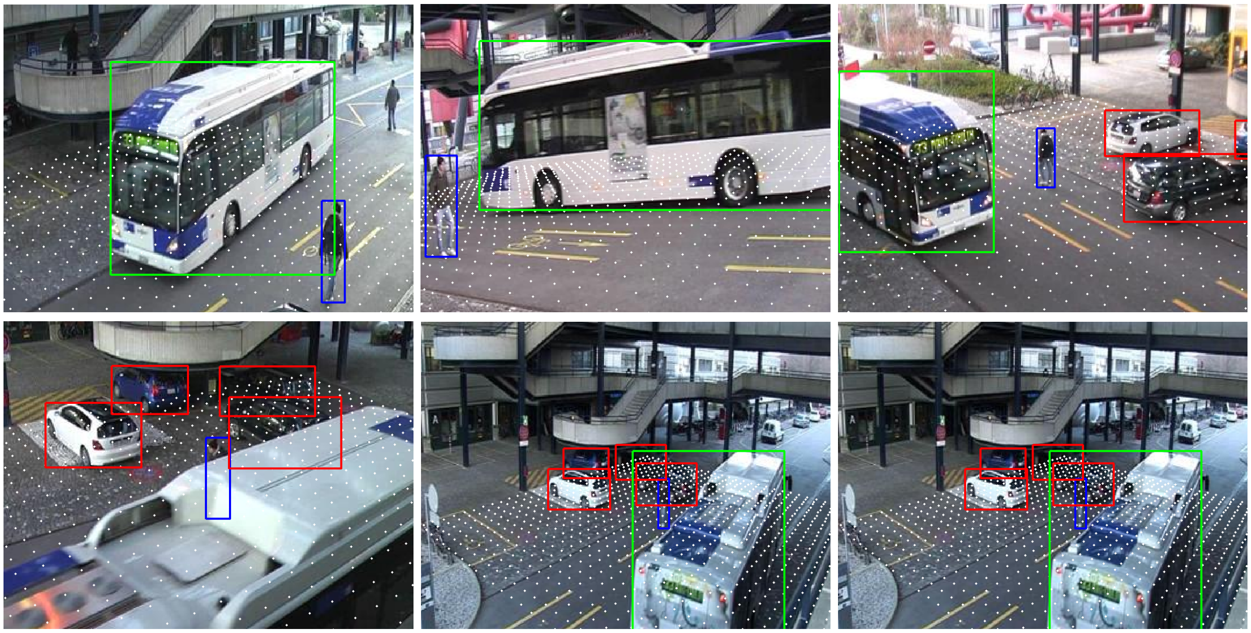


Figure 4. Frame taken from the 6 views of the dataset introduced in this paper. The ground plane is discretized in cells which center is indicated with a white point, and the target objects are surrounded by red bounding boxes for cars, green bounding boxes for buses and blue bounding boxes for people.

located on a first floor high, and the rest on a second floor to cover an area of 22m x 22m. The sequence was recorded at the EPFL university campus where there was a road with a bus stop, parking slots for cars and a pedestrian crossing. Afterwards, the most challenging 242 images from each view where selected. These frames contain different real situations where pedestrians, cars and buses appear and can cause high occlusions among them. Most of the taken images are uncorrelated on time, though there are some clusters of small sequences of images with time consistency. A total number of 1297 persons, 3553 cars and 56 buses were manually annotated with a bounding box around them. Although we aim at detecting buses, we do not evaluate them since the number of examples is too low. However, buses usually occlude big regions of the scene, and thus, it is important to include their detections. We divide the dataset in two splits for training and testing purposes (79 training images taken at different time steps than testing). In Figure 4 we show examples of the images of our dataset (available at *http://cvlab.epfl.ch/data/multiclass/*).

5.2. Implementation

Object detectors

For object detection we use Felzenszwalb *et al.* detector [8], which is based on mixture of deformable part models. We use the models provided by the authors, which the pedestrian's model is trained using INRIA Pedestrian dataset [3], and the cars and buses are trained using the training images of PASCAL VOC 2009 dataset [6].

For testing, we run each detector in all images and project all bounding boxes into the ground plane. We associate to each location of the ground plane the bounding box whose bottom center is closer to a certain distance and has the highest detection score among all possible, see Figure 2.

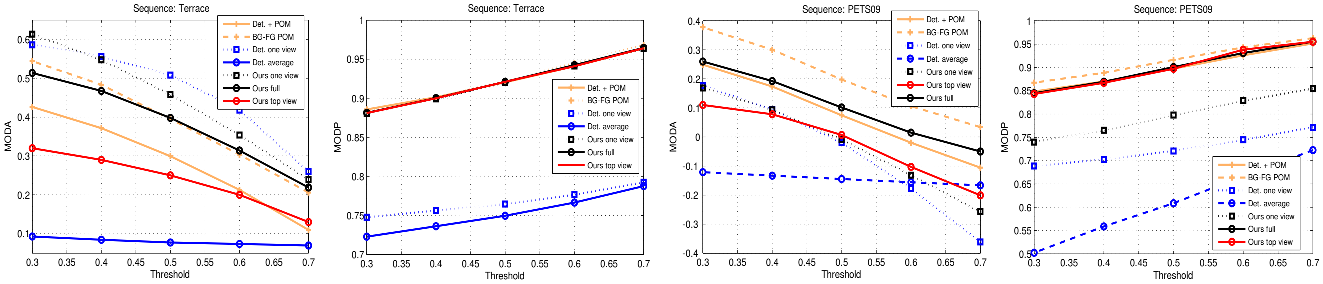
Inference

We use Algorithm 1 to approximately infer a solution. In all our experiments the iterative algorithm converged in less than 4 iterations. We use Belief Propagation (BP) [13] to infer the partial solutions of each iteration.

Learning

The parameters used in EPFL terrace sequence and PETS09 are obtained using one sequence for training and the other for testing in both cases. For the muti-object sequence we used the 79 training images. Parameters are learned using the max-margin paradigm as explained in Section 4.

The distance that determine the neighboring nodes $\mathcal{N}_i^v$ is set testing several distances and keeping the one that gives better performance for each dataset. To build the sets $\mathcal{K}_i^v$ we consider that a bounding box occludes another one when it is in front and it overlaps more than 0.7 of the area.



(a) (b) (c) (d)

Figure 5. MODA and MODP of EPFL terrace sequence [9] and PETS09 dataset of POM [9] comparing all baselines and related methods. Note that results of methods independently evaluated at each view (Det. one view and Ours one view) are only comparable between them, and not with methods that estimate the 2D location on the ground plane.

5.3. Results

We use several methods as baselines for multi-view detection to compare our method:

- *Averaging the detector of all views* (referred as Det. average): It consists of computing the average of the detection scores of all views, and performing a non-maximum suppression on the 2D ground plane.

- *CRF only using the top view* (Ours top view): We average the detector of all views, and we apply our model only on this generated top view. It does not include the occlusion maps.

- *POM with Background/Foreground Maps* (BG-FG POM)[3]: [9] is considered state-of-the-art on multi-camera pedestrian detection. However, we can only evaluate this method in the sequences with only one object class.

- *POM with Detection* (Det. + POM): In order to evaluate [9] when not using background subtraction, we generate the background/foreground binary masks with the output of detector after non-maxima supresion (NMS) in the image.

- *Detectors independently evaluated at each view* (Det. one view and Ours one view): We also compare the detector after NMS, and the CRF only using one single view as input. This problem is arguably different from locating the objects on the 2D ground plane, and hence, results are not comparable to multi-view detection methods.

For evaluation we use the standard metrics Multiple Object Detection Accuracy (MODA) and Multiple Object Detection Precision (MODP) following the protocol by [11].

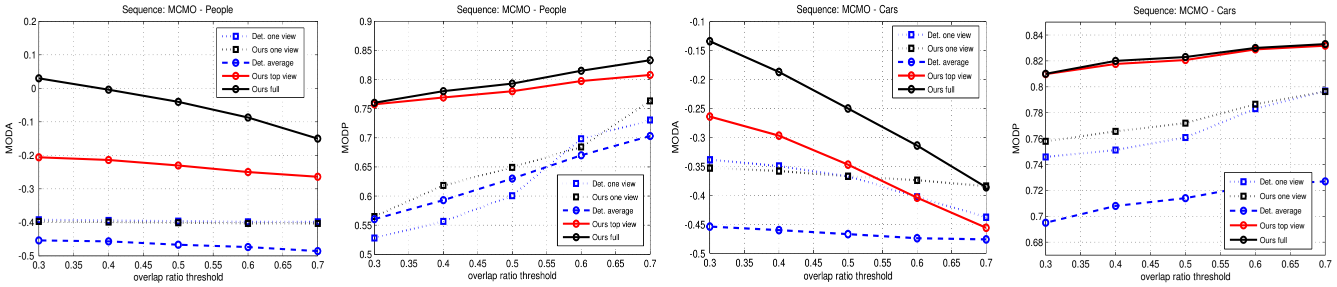
Terrace sequence. In Figures 5a and 5b MODA and MODP curves of the EPFL terrace sequence are depicted. Detection average baseline obtains a bad performance because averaging and taking the maximums on the ground plane fails especially when there are occlusions, or the detector for one of the view is unsuccessful. Our method obtain similar results as [9] when using background/foreground maps and better than using the simplified version of our model named Ours top view.

We observe that methods that evaluate the views independently obtain a different performance than taking into account all the views at the same time. This is mainly because these methods do not solve the same problem and are not comparable. Searching for the 2D location on the ground plane is a more challenging problem than looking for the bounding box on the image, because on the later it does not penalize not being consistent among views. For this reason, we plot the curves that take into account the views independently in dashes to emphasize that the methods evaluate different problems.

PETS09 S2/L1. We show MODA and MODP results on PETS09 dataset in Figure 5c and 5d respectively. In this sequence we observe the same behavior as in the EPFL terrace sequence. However, since in this sequence the monocular detector is less successful than in the terrace sequence, the method of [9] using background subtraction gets better performance because the background/foreground mask are more accurate. Compared to [9] using the output of the detector and NMS we perform better.

Multi-camera multi-object dataset. Results are shown in Figure 6. Our method substantially improves the baselines. We believe that this is because our method effectively handles occlusions. However, since in the dataset it appears objects at very different scales and viewpoints, and there are strong occlusions, the performance of the detector is relatively low compared to the other sequences. All previous methods are not suitable for this dataset because they are designed for one object class, or use background subtraction or tracking.

[3] code available at http://cvlab.epfl.ch/software/pom



(a) (b) (c) (d)

Figure 6. MODA and MODP of the multi-camera multi-object dataset in this paper on cars and persons object classes. Results of methods independently evaluated at each view (Det. one view and Ours one view) are not comparable to methods that estimate the 2D location on the ground plane. Buses are not evaluated but are included in the inference.

6. Conclusions

We proposed a novel approach to address the multi-object detection problem in a multi-camera environment. We use detection scores for each object, both to detect different object types and to be invariant to background changes. We model the system with a CRF that takes into account occlusions among objects within the same view, and consistency of the decision among different views. When background subtraction can be used, we obtain results that are similar to those of state-of-the-art methods. Furthermore, when it cannot be, we still get good results whereas these earlier methods become inapplicable. For future work we plan to incorporate into the system improved detectors that can handle partly occluded objects robustly.

Acknowledgments

This work was partially supported by the EU projects FP7-ICT-24314 Interactive Urban Robot, and FP7-ICT-248873 Robotic ADaptation to Humans Adapting to Robots. It was also supported in part by the Swiss National Science Foundation.

References

[1] A. Andriyenko and K. Schindler. Globally optimal multi-target tracking on a hexagonal lattice. In *ECCV*, 2010.

[2] J. Berclaz, F. Fleuret, and P. Fua. Principled detection-by-classification from multiple views. In *ICCVTA*, 2008.

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

[4] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for multi-class object layout. In *ICCV*, 2009.

[5] A. Ellis, A. Shahrokni, and J. M. Ferryman. Pets2009 and winter-pets 2009 results: A combined evaluation. In *Winter-PETS*, 2009.

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

[7] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 2010.

[8] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Discriminatively trained deformable part models, release 4. http://people.cs.uchicago.edu/ pff/latent-release4/.

[9] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-Camera People Tracking With a Probabilistic Occupancy Map. *PAMI*, 2008.

[10] T. Joachims, T. Finley, and C.-N. J. Yu. Cutting-plane training of structural svms. *ML*, 2009.

[11] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M.Boonstra, V. Korzhova, and J. Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *PAMI*, 2009.

[12] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 2006.

[13] F. R. Kschischang, B. J. Frey, and H. A. Loeliger. Factor graphs and the sum-product algorithm. *IT*, 2001.

[14] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

[15] A. Mittal and L. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. *IJCV*, 2003.

[16] V. I. Morariu and O. I. Camps. Modeling correspondences for multi-camera tracking using nonlinear manifold learning and target dynamics. In *CVPR*, 2006.

[17] K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *CVPR*, 2004.

[18] A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and H. Wensheng. Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *CVPR*, 2006.

[19] A. Sankaranarayanan, A. Veeraraghavan, and R. Chellappa. Object detection, tracking and recognition for multiple smart cameras. *Proceedings of the IEEE*, 2008.

[20] B. Taskar, C. Guestrin, and D. Koller. Max-margin markov networks. In *NIPS*, 2003.

[21] P. A. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 2004.

[22] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.