An Exemplar-based CRF for Multi-instance Object Segmentation

Xuming He
NICTA* and CECS, ANU, Canberra
xuming.he@nicta.com.au

Stephen Gould
CECS, ANU, Canberra
stephen.gould@anu.edu.au

Abstract

We address the problem of joint detection and segmentation of multiple object instances in an image, a key step towards scene understanding. Inspired by data-driven methods, we propose an exemplar-based approach to the task of instance segmentation, in which a set of reference image/shape masks is used to find multiple objects. We design a novel CRF framework that jointly models object appearance, shape deformation, and object occlusion. To tackle the challenging MAP inference problem, we derive an alternating procedure that interleaves object segmentation and shape/appearance adaptation. We evaluate our method on two datasets with instance labels and show promising results.

1. Introduction

Detection and localization of multiple objects in an image is one of the fundamental challenges of modern computer vision. The problem has become known as *multi-instance multi-class image segmentation* and is a key step towards understanding of the scene portrayed in the image. Amongst the many difficulties confronted when solving this challenging task is that interesting scenes often contain a high-degree of inter-object interaction, leading to large pose variation and occlusion (see Figure 1).

There have been many approaches that deal with multiple occluded objects in scenes. These can be divided into roughly two categories. In the first category bounding box object detectors are adapted to deal with occluded or missing parts [7, 15]. The limitation of these approaches is that they do not require the occlusion to be explained by another object. The second category of works treat multi-instance detection as a pixel labeling problem with smoothness priors [25]. Here each pixel is labeled with a class label and instance identifier. The occlusion is handled naturally as the discontinuity between two instances, but without long-range interactions these models struggle to correctly label the same object that is split into disconnected regions.

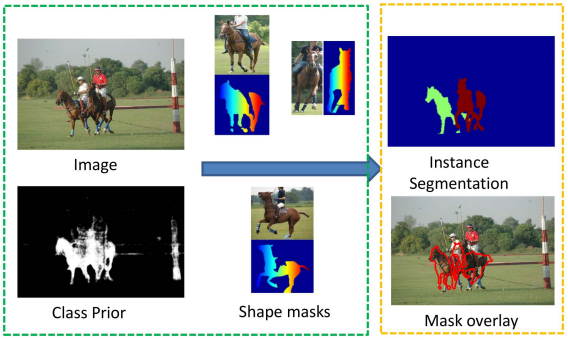

Figure 1. Illustration of our proposed approach on a horse instance segmentation task. The dashed green box indicates the input to our system, which includes a target image, the category-level prior and a set of reference shape masks from which the prior is derived. Our approach generates instance-level segmentations for all 'horse' objects and predicts their shape layouts, as shown in the dashed yellow box.

In this work we take a different approach that is motivated by the explosion in availability of annotated image data in recent years [21, 23]. Consequently, model-free approaches appear as strong contenders for ultimately solving the scene understanding problem [23, 29, 24, 9]. As a step in this direction, we propose an exemplar-based method for detecting and segmenting multiple interacting objects in a scene. Specifically, we consider the task of finding multiple instances of same-category objects in an image based on one or more reference images and their object shape masks. Figure 1 shows an example of this image parsing task, in which we obtain a pixel-level instance segmentation and shape layout based on a set of reference templates.

We formulate the joint detection and segmentation task as a multiclass (super-)pixel labeling problem, in which each (super-)pixel of a target image is assigned to an object instance label from an object candidate set. We design a conditional random field (CRF) that jointly models the object instance appearance, shape deformation, their activation and the inter-relation of occluding objects at the (super-)pixel level. In particular, we introduce a set

*NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the ARC through the ICT Centre of Excellence program.

of binary object variables and a series of potential functions, which impose (long-range) shape constraints on the (super-)pixel labels, label consistency between object and (super-)pixels, and smooth deformation of object shapes.

To parse an image, we compute the MAP estimate of the CRF model, which leads to a challenging energy minimization with hybrid variables. We propose an approximate inference procedure based on coordinate descent, which alternates between a segmentation step by (super-)pixel labeling and an instance learning step by optimizing shape mask and appearance models for each object.

Our approach has several key advantages. First, it does not require strong pre-learned object detectors, which allows it to be easily extended with new object categories by simply adding prototype images and corresponding masks. Nevertheless, our method is robust to moderate viewpoint/pose changes and appearance variation. Most important, however, is that our approach is robust to inter-object occlusion and is able to distinguish multiple overlapping object instances, as well as to group multiple disjoint image regions into objects. In addition, by running our method on a large dataset, we can incrementally discover new object exemplars along with their shape masks, which may lead to building a better object model.

We evaluate our method on two datasets with instance labels, one of which is a new segmentation dataset including more than 800 objects. We compare the performance of our approach with several baseline methods. The main contributions of our work are threefold: First, we propose an exemplar-based object instance segmentation framework; Second, we design a novel CRF model that jointly captures deformable object shape and occlusion between instances; Last, we introduce a new dataset with instance segmentation labels.

2. Related work

Being such an important problem in computer vision there have been numerous methods proposed for solving the multi-instance image segmentation problem and its variants. Here we discuss the works most related to ours.

Barinova et al. [3] addresses the problem of finding multiple object instances in natural and biological images based on the Hough voting paradigm. Unlike our work, they do not provide a pixelwise segmentation of the detected objects. Riemenschneider et al. [17] suggest integrating Hough voting with object support segmentation. However, they do not infer object shape and their deformation, nor do they have a unified CRF model.

Kuettel et al. [11] consider shape mask transfer from a training set for foreground object segmentation. However, they generate a single foreground segmentation, and do not distinguish co-occurring object instances (see also [18]). Similarly, Tighe and Lazebnik [24] integrates exemplar SVM-based mask transfer with region-based scene labeling for category-level segmentation. Kim and Xing [10] consider co-segmenting multiple foreground objects but do not infer object shape.

Other works use parts-based detector output to drive pixelwise segmentation. For example, Yang et al. [27] propose a model with explicit depth ordering of detected objects. This combined with a shape prior derived from a parts-based model [6] allows them to assign pixels to objects and thus segment the image. Wu and Nevatia [26] also use trained parts-based classifiers to detect objects and guide segmentation. Unlike Yang et al. [27] they do not assume a depth ordering but can still handle partial occlusions. Similarly Lin et al. [15] use a hierarchical parts-based model to detect and segment humans. In [5, 12], object detector outputs are integrated into a pixel-level CRF model to impose a soft top-down constraint. More recently, Ladicky et al. [13] integrates a part-based layout model with pixel-level labeling for human pose and layout estimation.

Unlike these methods, our model does not require pretrained models of objects or their parts. Instead our algorithm is data-driven, using a set of exemplars at test time to guide object detection and segmentation. Moreover, we allow the shape of our detections to deform based on appearance information rather than simply use the detections to inform segmentation.

Some works consider the problem from the other direction, using segmentation to inform bounding box object detection, e.g., [7]. However, their approach does not provide a pixelwise segmentation.

Perhaps most similar to our work is the layout consistent random field of Winn and Shotton [25]. In their approach a dense parts model is used to encode the shape of objects. Consistency in labeling is enforced via a Markov random field, which allows slight deformation of the parts. Unlike their approach, we do not assume a part-based object model, and it is easier for us to model deformable objects with multiple shape masks. In addition, Yao et al. [28] address holistic scene understanding with a CRF model similar to our work. The main difference is that we model object deformation and do not rely on object-specific detectors to generate proposals.

3. Modeling multiple instances

We consider parsing an image with multiple instances of some object class of interest to us. To tackle the problem, we assume a small set of reference images with object mask annotations is provided to represent the object appearances and shapes. Our goal is to segment every object instance in the target image and infer the layout of each instance with respect to the corresponding shape mask.

We formulate this multi-instance object segmentation as a scene labeling problem in which each image pixel is an-

notated by an object label and its shape mask. The object label groups together pixels into object instances, while the shape mask annotates the layout of each instance with respect to the reference masks and hence also identifies the object class.

Formally, assume we have a set of reference images $\{I_m^r\}_{m=1}^M$ and their corresponding object masks $\{S_m^r\}_{m=1}^M$. Based on the reference pairs, we generate a set of background and object instance hypotheses for a given target image I, denoted by $\mathcal{H} = \{h_0, h_1, \ldots, h_K\}$ where h_0 is the background. The details of hypothesis generation will be described in Section 4, and in the following, we assume that $\mathcal{H}$ is given.

We adopt a superpixel representation of the target image, and associate a label variable y_i with each superpixel in I, where $i \in \mathcal{V} = \{1, \ldots, N\}$. Here $\mathcal{V}$ denotes all the superpixel sites and N is their total number in the target image. The label y_i takes values from the object hypothesis set $\mathcal{H}$, and we assume no two objects occupy the same superpixel in the target image. We denote the location of the superpixel within the image by $\mathbf{x}_i$.

For each object hypothesis h_k, we introduce a binary variable o_k to indicate whether the hypothesis is active in the target image. The hypothesis h_k is represented by a mask parameter $\mathbf{s}_k$ and its appearance $\mathbf{a}_k$. The mask $\mathbf{s}_k$ is parametrized by a triplet $(m_k, \mathbf{c}_k, \mathbf{d}_k)$ where $m_k \in \{0, \ldots, M\}$ denotes the corresponding reference mask index, $\mathbf{c}_k$ the center position of the object instance, and $\mathbf{d}_k$ the mask deformation applied to $S_{m_k}^r$. Note that the m_k are fixed once the hypothesis set is generated. The background indicator o_B is always active, and its hypothesis has appearance parameter $\mathbf{a}_B$ only. Specifically, the background has no shape variable.

Our objective is to find an optimal labeling that interprets the target image with a small number of hypotheses. We achieve this by building a conditional Markov random field (CRF) on the superpixel label variables $\mathbf{Y} = \{y_i\}$, denoted as *superpixel variables*, and the object hypothesis variables $\mathbf{O} = \{o_k\}$, denoted as *object variables* and their parameters $(\mathbf{S}, \mathbf{A}) = \{(\mathbf{s}_k, \mathbf{a}_k)\}$. We connect each superpixel variable to its nearest neighbors in the image plane to encode a local smoothness constraint, and to all the object variables to represent the object level constraint. Specifically, let $\mathcal{N}$ be the superpixel neighborhood, we define an energy function E over $\mathbf{Y}$, $\mathbf{O}$, $\mathbf{S}$ and $\mathbf{A}$ with four types of potentials as follows.

$$E = \sum_{k=1}^{K} \psi_M(\mathbf{Y}, o_k) + \sum_{i=1}^{N} \sum_{k=0}^{K} \psi_d(y_i, \mathbf{s}_k, \mathbf{a}_k) \quad (1)$$
$$+ \sum_{i,j \in \mathcal{N}} \psi_s(y_i, y_j, \{\mathbf{s}_k\}) + \sum_{k=1}^{K} \psi_b(\mathbf{s}_k, \mathbf{a}_k),$$

where ψ_M encode the label configuration constraint between superpixels and object hypotheses, ψ_d are the global shape and appearance constraint per instance, ψ_s impose local rigidity/smoothness constraints for each object instance, and ψ_b are the bias term for the mask and appearance parameters. The details of each term will be described in the following subsections, and a graphical representation of the model is shown in Figure 2.

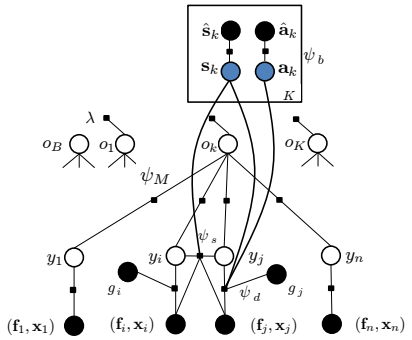

Figure 2. Factor graph representation of our model. Black nodes are observed variables and blue nodes are instance parameters.

3.1. Label consistency and sparsity

We require that the superpixel labeling be consistent with the active hypotheses: if any patch y_i takes value k, then the hypothesis indicator $o_k = 1$; otherwise $o_i = 0$. Such constraints can be encoded by the following potential:

$$\psi_M(\mathbf{Y}, o_k) = \sum_{i=1}^{n} [\![y_i = k]\!][\![o_k = 0]\!] W + \lambda [\![o_k = 1]\!] \quad (2)$$

where W is a large positive constant that penalizes any label inconsistency between the superpixel and object variables. The positive constant λ is the cost for being an active hypothesis, and encourages a sparse set of object instances being instantiated in the target image.

3.2. Object shape and appearance

Each active object hypothesis imposes a global shape and appearance constraint based on the reference mask/image, which is represented by ψ_d. We include both the geometric and appearance relationship as follows:

$$\psi_d(y_i, \mathbf{s}_k, \mathbf{a}_k) = \Big(-\alpha_0 \log\big(S_{m_k}(\mathbf{x}_i - \mathbf{c}_k - \mathbf{d}_{ki})\big) \quad (3)$$
$$- \alpha_1 \log(g_{ik}) + \phi_a(\mathbf{f}_i, \mathbf{a}_k) + \alpha_2 \phi_c(\mathbf{s}_k) \Big) [\![y_i = k]\!]$$

where $\mathbf{f}_i$ is a local superpixel feature vector, g_{ik} is the category prior and $\{\alpha_i\}_{i=0}^3$ are the weighting coefficients for the unary terms. Here $\mathbf{x}_i$ denotes the image position of superpixel i, and $\mathbf{d}_{ki}$ is the average shape deformation of the kth instance on the ith superpixel. We define an appearance

cost $\phi_a(\mathbf{f}, \mathbf{a})$ for mismatch between the superpixel appearance feature and the object appearance, and a contour cost $\phi_c(\mathbf{s}_k)$ for misalignment of shape mask and image edges.

To compute the appearance cost, we first build an instance specific color model for each hypothesis. We represent each superpixel by its mean color in the CIE Lab space, and learn a Gaussian Mixture Model, denoted by $p_{\text{GMM}}(\mathbf{f}; \mathbf{a}_k)$, for the kth hypothesis. The appearance cost is defined by the negative log likelihood of the superpixel color feature, i.e., $\phi_a(\mathbf{f}_i, \mathbf{a}_k) = -\log(p_{\text{GMM}}(\mathbf{f}_i, \mathbf{a}_k))$. The contour cost is computed by mapping the shape mask contour into the image and estimating the oriented Chamfer distance [22] between the mask contour and local image edges.

The first term is a mask cost that constrains the scope of the objects. We allow slight mismatches between the shape templates and objects in the image by blurring the binary mask with a Gaussian filter with a kernel width of 10% mask height. The mask cost for the ith superpixel is computed by mapping the pixel-wise soft mask onto the superpixel and taking its average, which also takes into account the object center $\mathbf{c}_k$ and the deformation $\mathbf{d}_{ki}$.

We further incorporate the object category prior into the energy function. The category prior can be obtained by any scene labeling method (e.g, [9]) that generates a marginal probability distribution of the categories for each superpixel. The category prior g_{ik} is defined by the object category probability p_i^c if $k > 0$, and $1 - p_i^c$ if $k = 0$.

3.3. Local rigidity and smoothness of deformation

We assume the shape deformation of each object instance is small with respect to the reference masks. We first consider a local rigidity constraint on any two neighboring superpixels of an object instance, requiring that the spatial distance between the two superpixels keeps approximately constant. Let the two neighboring sites be i and j (i.e., $(i, j) \in \mathcal{N}$), we define the energy cost $\psi_s(\cdot)$ as follows,

$$\psi_s(y_i, \mathbf{s}_i, y_j, \mathbf{s}_j) \tag{4}$$
$$= \beta \cdot \begin{cases} \frac{1}{\|\mathbf{x}_i - \mathbf{x}_j\|^2} \|\mathbf{d}_{ki} - \mathbf{d}_{kj}\|^2, & y_i = y_j = k \\ \gamma(1 - e(\mathbf{f}_i, \mathbf{f}_j)), & y_i \neq y_j \end{cases}$$

where β is the weighting coefficient for the local rigidity term, $e(\mathbf{f}_i, \mathbf{f}_j)$ is the local object boundary probability, γ is a coefficient modulating the boundary cost.

The local object boundary probability $e(\mathbf{f}_i, \mathbf{f}_j)$ is estimated from region boundary cues. We compute the global Pb value Pb_g [2], and take the average value along the boundary between two superpixels as the probability.

3.4. Shape and appearance bias

The generated object hypothesis set for a target image provides an initial estimation of each object instance's shape and appearance. We denote those parameters of the kth instance as $\hat{\mathbf{s}}_k = (\hat{\mathbf{c}}_k, \hat{\mathbf{d}}_k)$ and $\hat{\mathbf{a}}_k$ (see Sec. 4 for details). The shape and appearance bias term uses these initial estimates as a prior and requires that the object instance parameters $(\mathbf{s}_k, \mathbf{a}_k)$ do not deviate too much from them. More concretely, we define the potential ψ_b as follows:

$$\psi_b(\mathbf{s}_k, \mathbf{a}_k) = \sigma_d \|\mathbf{s}_k - \hat{\mathbf{s}}_k\|^2 + \sigma_a \|\mathbf{a}_k - \hat{\mathbf{a}}_k\|^2 \tag{5}$$

where σ_d and σ_a are the weighting coefficients for the shape deformation and appearance constraints, respectively. By this potential function, we enforce that the appearance of an object instance remains unchanged while the shape mask can adapt to the target image. This partially prevents blending of two neighboring instances sharing similar color.

Overall, our model can be viewed as a random field capturing both local smoothness of object deformation and long-range dependency between object parts. The layer of superpixel label variables explicitly models the local deformation of object shapes, and transition between object instances at their boundaries. The object-level dependency is induced by the shape and appearance components, which encourages grouping geometrically consistent patches into individual object instance.

4. Model inference

Our approach requires a hypothesis generation process to provide the object hypothesis set $\mathcal{H}$. While this initialization is not a core component in our model, it affects the efficiency and performance of the approximate inference in our approach. In this section, we first introduce our initialization step, followed by the overall inference algorithms. We briefly describe the parameter tuning method used in our system at the end.

4.1. Hypothesis generation

We initialize the object hypothesis set in two stages. The first stage uses a modified Hough voting scheme [3] and an exemplar SVM [16] to propose the location and scale of candidate object instances. We denote the initial estimation of object centers as $\mathcal{C} = \{\hat{\mathbf{c}}_k\}$. For each hypothesis, we rescale the corresponding reference image and mask to match the scale of the proposed instance.

More specifically, our modified Hough voting method comprises two enhancements. First, we incorporate category information by weighting the voting score with the foreground class probability. This eliminates many weak hypotheses and better estimates object scales and centers. Second, we adopted the generalized PatchMatch algorithm [4] to efficiently compute a patch-wise matching from the target image to every reference image. We use HOG features to account for color variations. The exemplar SVMs are trained with reference images only and the reference masks are used to remove the background clutter.

The second stage initializes the object deformation and appearance models. We first estimate a dense support of each object hypothesis on the image plane. Our strategy is to define a set of seeds for the object, which consists of all the matches to the valid region of reference images at the PatchMatch step. We also estimate background seeds from the foreground class probability map. Given the foreground and background seeds, we run GrabCut [19] to obtain an initial dense support for each object hypothesis.

We use the superpixel color features in the dense region support to build the initial appearance model based on GMMs, i.e., $\{\hat{\mathbf{a}}_k\}_{k=1}^K$. Note that we share the Gaussian components between all the instance models and represent each object's appearance by the mixture coefficients. The background model can be generated similarly from the category prior. In each object hypothesis, we define the initial deformation by the flow derived from the PatchMatch.

The hypothesis generation step constrains the possible number of object instances in the later search stage, while the object instance's shape parameters will still be updated. Our method can generate a set of initial object hypotheses with good quality, which speeds up the subsequent inference step.

4.2. Joint inference with alternating procedure

Given an image, our goal is to parse it into foreground object instances (from a category) and background. We achieve this by minimizing the energy function $E(\mathbf{Y}, \mathbf{O}, \mathbf{S}, \mathbf{A})$, in which our inference algorithm searches for the optimal configuration of object and pixel labels $(\mathbf{Y}^\star, \mathbf{O}^\star)$ and estimates the shape and appearance of all instances $(\mathbf{S}^\star, \mathbf{A}^\star)$. Note that our solution not only segments individual object instances, but also provides a layout interpretation of each object based on the estimated shape mask.

However, this is a challenging optimization task as we have a hybrid objective function with both discrete and continuous variables. To efficiently minimize the energy function, we adopt a coordinate descent strategy that solves two simpler sub-problems in an alternating way. More specifically, we decompose the joint minimization into one discrete and one continuous problem. First, we fix the object shape and appearance parameters and infer the object and superpixel variables. Then given the object and superpixel labels, we adjust the shape and appearance parameters of active object instances. Mathematically, at iteration t, we have the following updates

$$(\mathbf{Y}^t, \mathbf{O}^t) = \underset{\mathbf{Y},\mathbf{O}}{\operatorname{argmin}} E(\mathbf{Y}, \mathbf{O}, \mathbf{S}^{t-1}, \mathbf{A}^{t-1}), \tag{6}$$

$$(\mathbf{S}^t, \mathbf{A}^t) = \underset{\mathbf{S},\mathbf{A}}{\operatorname{argmin}} E(\mathbf{Y}^t, \mathbf{O}^t, \mathbf{S}, \mathbf{A}), \tag{7}$$

In the following, we will describe our algorithms for solving Equations 6 and 7.

A. Inference for pixel and object labels

We first address the discrete optimization sub-problem in Equation 6. This reduces to an energy minimization for a multilabel pairwise Markov random field. We adopt a move-making approach that searches for the optimal α-expansion move to decrease the energy function. However, due to the structure of our energy function (e.g., Equation 4), the binary sub-problems in α-expansion are not submodular. We resort to QBPO to solve minimization in the expansion moves [20]. As we observed in our experiments, the QPBO subroutine usually yields integral solutions. In case the non-integral solutions are generated from the α-expansion, we round the solution according to the shape mask support of the current object hypothesis.

B. Shape and appearance update

The continuous optimization in Equation 7 reduces to finding the optimal shape parameters $\mathbf{S}$ and the appearance parameters separately. However, it is also nontrivial to solve the shape optimization as the objective function is non-convex. We choose to discretize the object center and deformation space, and convert this into a multilabel energy minimization problem. We solve this discrete problem by multi-start Iterated Conditional Modes (ICM). For the appearance model update, we simply use (a few steps of) gradient descent. In practice, we only compute them every few iterations to slowly update these variables.

4.3. Parameter estimation

In this work, we focus on the model inference and employ a simple greedy strategy to estimate the model parameter. We first manually set $W = 10^5$ for the inconsistency penalty. For other parameters, we sequentially search for their values based on a small training dataset and leave-one-out cross-validation. For each parameter, we do a grid search at 5 values (empirically selected).

5. Experiments

We evaluate our method on two datasets with pixel-wise instance labels and partial occlusions. The first dataset is derived from an existing scene labeling dataset—the Polo dataset [29, 9]. For comparison, we also test our method on the TUD Crossing dataset [17], which consists of a group of 3 to 10 pedestrians per image.

Note that the original Polo dataset has pixel-wise category labeling only. We augment the original labeling by additional instance segmentation labels. In particular, we select the *horse* category, and manually generate all instance-level segmentations. We choose this dataset for two main reasons: First, the dataset includes many scenes with multiple object instances interacting with and occluding each other. In addition, the main foreground object categories

Figure 3. Examples of reference images and shape masks selected from the Polo and TUD dataset. They covers a diverse set of viewpoints and the masks are color-coded to visualize their layout.

have deformable shapes and a variety of poses. Both factors make the segmentation task very challenging. Some examples of the ground truth labelling is shown in Figure 5.

5.1. Experimental setup

Polo Dataset. We follow the setting in [9], which splits the whole dataset into 80 training images and 237 test images. Based on the label transfer procedure in [9], we train a 6-category classifier from the training set and generate the category-level prior for the test set.

We use the SLIC method [1] to compute a superpixel representation of the test images tuned to yield about 1500 superpixels per image. We map pixel-level features onto superpixels by averaging over each superpixel.

We select ten templates from the subset of training images that include only one object instance. This makes it straightforward to obtain the object mask from the category-level labeling. Some templates are slightly incomplete due to occlusion. However, our method is robust to these errors. The selected templates cover about six or seven typical viewpoints and a variety of poses. Some of the reference templates are shown in Figure 3. The masks are color-coded so that its layout is easy to see. Note that we can easily convert between left and right oriented objects.

We initialize the object hypothesis set by generating a number of initial hypotheses ranging from 30 to 50 per image. For each object instance, we build a Gaussian mixture model with at most 15 components for its color appearance.
TUD Pedestrian. We use the same setting as above for superpixels and build Gaussian mixture models with 20 components. For the category-level prior, we train a foreground/background pixel labeling model based on the Stanford dataset [8] and use the foreground probability. We select eight templates from the TUD Pedestrian training dataset and also include their mirror version. Some examples are shown in Figure 3.
Baseline methods. We build two baseline methods for quantitative comparison. The first baseline, denoted as 'E-SVM', generates instance segmentation from the exemplar SVMs trained with our templates, and transfers the masks to the detected object instances. The second baseline is based on the initialization of our method. Instead of over-generating hypotheses, we assume the number of objects in each image is known giving the baselines an unfair advantage. For both methods we select this number of hypotheses with the highest voting score. Given the object center and mask information, we run the same GrabCut procedure to obtain the object instance segmentation. This baseline is referred to as 'HV+GC'. We start from the strongest hypothesis and greedily generate all the instance labeling.
Running time. Our matlab implementation takes roughly 10 minutes for feature extraction, 1 minute for proposal generation and 5 minutes for inference per image on a desktop with a 3.0GHz Intel Core-i7 CPU and 24 Gb memory.

Method	Mi-AP	Mi-AR	Ma-AP	Ma-AR	Avg-FP	Avg-FN
E-SVM	38.5	33.6	43.9	38.3	1.0	1.3
HV+GC	44.6	38.7	**61.7**	49.4	0.6	**0.7**
Ours-S	49.9	53.2	57.6	68.7	0.5	0.8
Ours+S	**50.9**	**53.7**	57.4	**68.8**	**0.4**	0.8

Table 1. Performance comparison on Polo horse dataset. 'E-SVM' and 'HV+GC' are two baseline methods, and 'Ours-S' and 'Ours+S' are our results without and with shape deformation respectively. See the text for details.

5.2. Segmentation performance

Metrics for segmentation. Given a test image, our method predicts a superpixel-wise instance labeling, which is mapped back to the pixel level. To compare with the ground-truth labeling, we search for the matching pairs between the prediction and the ground-truth. This can be formulated as a bipartite matching problem in which the matching profit is measured by the area of overlap. All the matching pairs are found by maximizing the overall profit. Once the matching pairs are found, we compute precision and recall rates. Note that since we take the MAP estimate of the object segmentation, we only have a single point on the PR curve.

We calculate three sets of evaluation metrics for the whole dataset: 1) Pixel-wise precision rate per object (Mi-AP), which is averaged over all object predictions; and pixel-wise recall rate per object (Mi-AR), which is averaged over all groundtruth objects. 2) Overall pixel-wise precision rate (Ma-AP) and recall rate (Ma-AR), which are averaged over all the pixels. 3) Detection metric: average false positives per image (Avg-FP) and average miss detections per image (Avg-FN). Here, due to heavy occlusion, we use an optimistic criterion with overlap threshold of at least one pixel to determine whether an object is hit or missed.
Polo Dataset. We summarize our results in the Table 1. Two settings of our model are evaluated. In the first setting, we do not update the shape information, denoted as 'Ours-S' and the second setting is our full model, denoted as 'Ours+S'. We can see from the results, because we model the segmentation at the object instance level, both versions of our model perform significantly better than the two baselines in terms of micro average precision and recall. For macro average precision, the baseline 'HV+GC' has a higher score, which is not surprising since it assumes

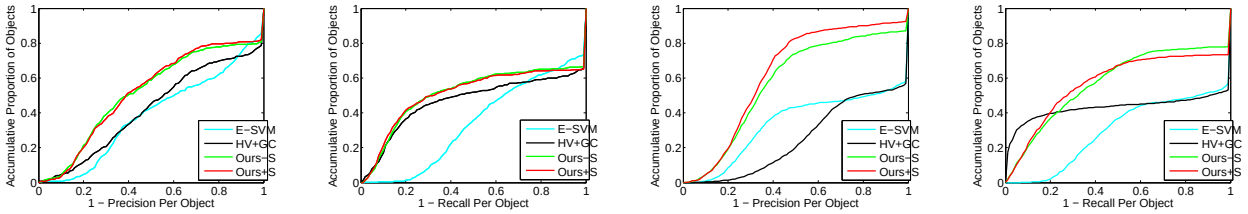

Figure 4. Comparison of accumulative precision and recall distribution of the baselines and our method. **Left: Polo dataset. Right: TUD dataset.** Our method's results concentrate towards higher precision/recall rate.

Method	Mi-AP	Mi-AR	Ma-AP	Ma-AR	Avg-FP	Avg-FN
E-SVM	33.7	29.5	49.5	33.0	2.3	2.4
HV+GC	24.9	42.9	41.6	51.9	2.3	2.4
Ours-S	57.5	**57.4**	61.2	63.5	0.7	**1.3**
Ours+S	**62.6**	56.9	**64.8**	**64.5**	**0.3**	1.5

Table 2. Performance comparison on TUD pedestrian dataset. See the text for details.

knowing the right number of instances. And it is at the price of having much lower macro average recall. On the other hand, it is interesting to see 'HV+GC' has a lower average miss detection rate.

More detailed difference between 'HV+GC' and 'Ours+S' can be seen from the accumulative distribution of the precision/recall rate per object in Figure 4. Our approach has more detections with higher precision and recall rate, while the baselines are more uniformly distributed.

We show some examples of our results on the Polo dataset in Figure 5. Those examples include multiple objects with large pose variation, mutual occlusion and novel poses. Our method seems to be able to handle challenging scenarios and achieve good instance level segmentation. We also show the overlay of the shape mark on the objects and their shape layout. Although sometimes the masks are not closely aligned with object boundaries, the inferred masks and layouts are largely correct.

TUD Pedestrian. The results on the TUD crossing pedestrian dataset are summarized in Table 2 and Figure 4. We can see the same trend as in the Polo dataset. We also show some of our results in Figure 6. Comparing with [17], our performance is on par with the state of the art qualitatively.

5.3. Shape mask discovery

Our method can be used to discover new shape masks in a large dataset [14]. In Figure 7, we show a set of example shape masks that our model found from the test dataset. They are similar to the original 10 shape masks but also differs in some aspects. By accumulating different shape and their layouts, our approach can be employed to learn a better object detection model with weaker supervision. To verify this, we add 20 more horse templates generated from our algorithms and re-train an exemplar SVM with a total of 30 examples. Then we re-evaluate the E-SVM baseline and achieve 43.7% Mi-AP, 38.8% Mi-AR, 45.5% Ma-AP, and 41.0% Ma-AR, which is significantly better than the hand-labeled 10-template setting.

Figure 6. Example results on TUD dataset. Left: input images; Middle: segmentation results; Right: overlay with template masks.

Figure 7. Examples of new shapes found by our method.

6. Conclusion

In this paper, we present an exemplar-based approach to multiple instance segmentation, focusing on the challenging problem of large pose variation and object occlusion. We describe a CRF model that jointly captures object instance shape, appearance and their occlusion and propose an efficient alternating algorithm to solve the MAP inference. Our method is evaluated on new and existing datasets with pixel-wise instance labeling, and the results demonstrate the effectiveness of the proposed approach in comparison with two baselines and qualitatively against the state of the art.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels. *EPFL, TR*, 2010. 6

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33(5):898–916, 2011. 4

[3] O. Barinova, V. Lempitsky, and P. Kohli. On detection of multiple object instances using hough transforms. *PAMI*, 2012. 2, 4

[4] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized PatchMatch correspondence algorithm. In *ECCV*, 2010. 4

[5] T. Brox, L. Bourdev, S. Maji, and J. Malik. Object segmentation by alignment of poselet activations to image contours. In *CVPR*, 2011. 2

Figure 5. Examples of instance segmentation generated by our model on the Polo dataset. Note that color is only used to distinguish different objects. Best viewed on screen.

[6] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE PAMI*, 2010. 2

[7] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. In *CVPR*, 2011. 1, 2

[8] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *ICCV*, 2009. 6

[9] S. Gould and Y. Zhang. Patchmatchgraph: building a graph of dense patch correspondences for label transfer. In *ECCV*. 2012. 1, 4, 5, 6

[10] G. Kim and E. P. Xing. On multiple foreground cosegmentation. In *CVPR*, 2012. 2

[11] D. Kuettel and V. Ferrari. Figure-ground segmentation by transferring window masks. In *CVPR*, 2012. 2

[12] L. Ladickỳ, P. Sturgess, K. Alahari, C. Russell, and P. H. Torr. What, where and how many? combining object detectors and crfs. In *ECCV*, 2010. 2

[13] L. Ladicky, P. H. S. Torr, and A. Zisserman. Human pose estimation using a joint pixel-wise and part-wise formulation. In *CVPR*, 2013. 2

[14] Y. J. Lee and K. Grauman. Shape discovery from unlabeled image collections. In *CVPR*, 2009. 7

[15] Z. Linand, L. S. Davis, D. Doermann, and D. DeMenthon. Hierarchical part-template matching for human detection and segmentation. In *ICCV*, 2007. 1, 2

[16] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, 2011. 4

[17] H. Riemenschneider, S. Sternig, M. Donoser, P. M. Roth, and H. Bischof. Hough regions for joining instance localization and segmentation. In *ECCV*. 2012. 2, 5, 7

[18] A. Rosenfeld and D. Weinshall. Extracting foreground masks towards object recognition. In *ICCV*, 2011. 2

[19] C. Rother, V. Kolmogorov, and A. Blake. GrabCut: Interactive foreground extraction using iterated graph cuts. In *SIGGRAPH*, 2004. 5

[20] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *CVPR*, 2007. 5

[21] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2008. 1

[22] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *ICCV*, 2005. 4

[23] J. Tighe and S. Lazebnik. SuperParsing: Scalable nonparametric image parsing with superpixels. In *ECCV*, 2010. 1

[24] J. Tighe and S. Lazebnik. Finding things: Image parsing with regions and per-exemplar detectors. In *CVPR*, 2013. 1, 2

[25] J. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *CVPR*, 2006. 1, 2

[26] B. Wu and R. Nevatia. Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. *IJCV*, 82(2):185–204, 2009. 2

[27] Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes. Layered object detection for multi-class segmentation. In *CVPR*, 2010. 2

[28] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*, 2012. 2

[29] H. Zhang and L. Quan. Partial similarity based nonparametric scene parsing in certain environment. In *CVPR*, 2011. 1, 5