# Multi-instance Object Segmentation with Exemplars

Xuming He
Computer Vision Group
NICTA and ANU
xuming.he@nicta.com.au

Stephen Gould
Research School of Computer Science
Australian National University
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 multi-instance segmentation using a small set of annotated reference images. We design a novel CRF model that jointly models object appearance, shape deformation, and object occlusion at the superpixel level. To tackle the challenging MAP inference problem, we derive an alternating procedure that interleaves object segmentation and layout adaptation.*

## 1. Introduction

Detection and segmentation of multiple objects, one of the fundamental challenges in computer vision, is a key step towards scene understanding. Amongst the many difficulties that need to be addressed when solving this task is that interesting scenes often contain a high-degree of inter-object interaction, leading to large pose variation and occlusion. Furthermore, occlusion boundaries between objects are often weak, especially for objects of the same class.

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 [2, 11]. 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, such as the layout consistent CRFs [10]. 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 in two (or more) disconnected regions.

In this work we take a different approach that is motivated by the explosion in availability of annotated image data and the model-free approaches in recent years [8]. We propose an exemplar-based method for detecting and segmenting multiple interacting objects in a scene. Our key idea is to localize and segment multiple object instances based on one or more reference images and corresponding shape masks.

We formulate this joint detection and segmentation as a multiclass (super-)pixel labeling problem, in which each (super-)pixel of a target image is assigned to an object instance label. We design a conditional Markov random field (CRF) that jointly models the appearance and shape deformation of each object instance together with the inter-relation of occluding objects at the (super-)pixel level. To parse an image, we compute the MAP estimate of the CRF model. However, this leads to a challenging energy minimization with both discrete and continuous 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 object shape mask and appearance model.

Our approach has several key advantages. First, it does not require pre-learned models of object detectors [12], 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.

We introduce a new segmentation dataset with object instance labels, which includes more than 800 objects. We evaluate our method on this dataset and compare its performance with two baseline methods.

**Related work.** Barinova et al. [1] addresses the problem of finding multiple object instances in natural and biological images. Unlike our work, they do not provide a pixelwise segmentation of the detected objects. Riemenschneider et al. [5] 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. [4] consider transfer shape masks from a training set for foreground object segmentation. However, they generate a single foreground segmentation, and do not distinguish co-occurred object instances. Yao et al. [13] 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.

## 2. Modeling multiple instances

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 $\mathcal{H} = \{h_0, h_1, \cdots, h_K\}$ where $h_0$ is the background. The details of hypothesis generation will be described in Section 3. For now we assume $\mathcal{H}$ is given.

We adopt a superpixel representation of the target image, and associate a label variable $y_i$ to 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 $\mathcal{H}$. 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 $\mathbf{s}_k$ and 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 $S_{m_k}^r$. The background indicator $o_0$ is always active, and its hypothesis has appearance parameter $\mathbf{a}_0$ only.

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 variable $\mathbf{O} = \{o_k\}$, denoted as *object variables*, and their associated parameters $(\mathbf{S}, \mathbf{A}) = \{(\mathbf{s}_k, \mathbf{a}_k)\}$. We connect each superpixel variable to its spatial 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 $\psi_M$ encode the label configuration constraint between superpixels and object hypotheses, $\psi_d$ are the global shape and appearance constraint per instance, $\psi_s$ impose local rigidity/smoothness constraints for each object instance, and $\psi_b$ are the bias terms for the mask and appearance parameters. Our model is depicted graphically in Figure 1.

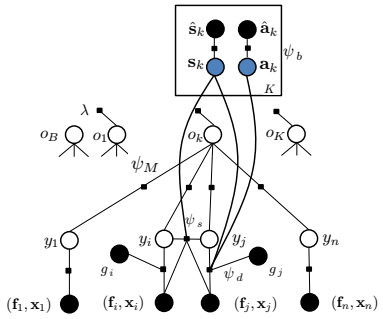


Figure 1. Factor graph representation of our model. Black nodes are observed variables; blue nodes represent instance parameters.

### 2.1. Label consistency and sparsity

We require that the superpixel labeling be consistent with the active hypotheses, which is encoded by the energy term:

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

where $W$ is a large positive constant that penalizes any label inconsistency between the superpixel and object variables. The positive $\lambda$ is the cost for being an active hypothesis.

### 2.2. Object shape and appearance

Each active object hypothesis (i.e., $o_k = 1$) imposes a global shape and appearance constraint based on the reference image/mask pairs:

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

where $\mathbf{f}_i$ is a local superpixel feature vector, $g_{ik}$ is the category prior and $\alpha$ is the weighting coefficient for the prior term. Here $\mathbf{x}_i$ denotes the image position (i.e., centroid) of superpixel $i$, and $\mathbf{d}_{ki}$ is the average shape deformation of the $k$th instance on the $i$th superpixel. We define an appearance cost $\phi_a(\mathbf{f}, \mathbf{a})$ for mismatch between the superpixel appearance feature and the object appearance.

To compute the appearance cost, we first build an instance specific color model for each hypothesis. We learn a color Gaussian Mixture Model, denoted by $p_{\text{GMM}}(\mathbf{f}; \mathbf{a}_k)$, for the $k$th hypothesis. The appearance cost is then defined by $\phi_a(\mathbf{f}_i, \mathbf{a}_k) = -\log(p_{\text{GMM}}(\mathbf{f}_i, \mathbf{a}_k))$. The first term in Equation 3 is a mask cost that constrains the scope of the

objects. The mask cost for the $i$th superpixel is computed by mapping the pixel-wise mask onto the superpixel and taking its average, which also takes into account the object center $\mathbf{c}_k$ and amount of deformation $\mathbf{d}_{ki}$.

We further incorporate into 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 variable $g_{ik}$ is defined by the object category probability $p_i^c$ if $k > 0$, and $1 - p_i^c$ if $k = 0$.

### 2.3. Local rigidity and smoothness of deformation

We assume the shape deformation of each object instance is small with respect to the reference masks. Let $i$ and $j$ be two neighboring sites in the target image, i.e., $(i, j) \in \mathcal{N}$. Then we define the energy cost $\psi_s(\cdot)$ as,

$$\begin{aligned}&\psi_s(y_i, \mathbf{s}_i, y_j, \mathbf{s}_j) \qquad (4)\\ &= \beta \cdot \begin{cases} \frac{1}{\|\mathbf{x}_i - \mathbf{x}_j\|^2} \|\mathbf{d}_{ki} - \mathbf{d}_{kj}\|^2, & y_i = y_j = k > 0 \\ \gamma(1 - e(\mathbf{f}_i, \mathbf{f}_j)), & y_i \neq y_j \\ \epsilon_p, & y_i = y_j = 0 \end{cases}\end{aligned}$$

where $\beta$ is the weighting coefficient for the local rigidity term, $e(\mathbf{f}_i, \mathbf{f}_j)$ is the local object boundary probability, $\gamma$ is a coefficient modulating the boundary cost, and $\epsilon_p$ is the constant cost for both being background.

### 2.4. Shape and appearance bias

The object hypothesis set $\mathcal{H}$ for a target image provides an initial estimation of each object instance's shape and appearance. We denote such parameters of the $k$th instance as $\hat{\mathbf{s}}_k = (\hat{\mathbf{c}}_k, \hat{\mathbf{d}}_k)$ and $\hat{\mathbf{a}}_k$. The shape and appearance bias term uses these initial estimates as a prior:

$$\psi_b(\mathbf{s}_k, \mathbf{a}_k) = \sigma_d ||\mathbf{s}_k - \hat{\mathbf{s}}_k||^2 + [\![\mathbf{a}_k = \hat{\mathbf{a}}_k]\!] W \qquad (5)$$

where $\sigma_d$ is the weighting coefficient for the shape deformation constraints, and $W$ is a large constant cost.

## 3. Model inference and learning

### 3.1. Hypothesis generation

We initialize the object hypothesis set in two stages. In the first stage, we estimate the scale and center locations of object hypotheses using the set of object templates and the Hough voting method [1]. The second stage initializes the object deformation and appearance models. Here we estimate a dense support of each object hypothesis on the image plane. Our strategy is to define a set of seeds for the object by using its sparse support, which consists of all the voters for the corresponding Hough mode. We also have a good estimate of background from the reference mask. Given the foreground and background seeds, we run a GrabCut-like algorithm [6] to obtain an initial dense support for each object hypothesis.

### 3.2. Joint inference with alternating procedure

We parse an image 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 $(\mathbf{S}^\star, \mathbf{A}^\star)$.

However, this is a challenging optimization task as we have a hybrid objective function with both discrete and continuous variables. 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

$$(\mathbf{Y}^t, \mathbf{O}^t) = \arg\min_{\mathbf{Y},\mathbf{O}} E(\mathbf{Y}, \mathbf{O}, \mathbf{S}^{t-1}, \mathbf{A}^{t-1}), \qquad (6)$$

$$(\mathbf{S}^*, \mathbf{A}^*) = \arg\min_{\mathbf{S},\mathbf{A}} E(\mathbf{Y}^t, \mathbf{O}^t, \mathbf{S}, \mathbf{A}), \qquad (7)$$

$$(\mathbf{S}^t, \mathbf{A}^t) = \left((1 - \eta)(\mathbf{S}^{t-1}, \mathbf{A}^{t-1}) + \eta(\mathbf{S}^*, \mathbf{A}^*)\right). \qquad (8)$$

where $\eta$ controls the updating rate of the instance parameters. We discretize the continuous sub-problem and use Graphcut [7] to solve both minimization tasks.

### 3.3. Parameter estimation

We fix $W = 10^5$ for the inconsistency penalty and $\epsilon_p = 0.1$ for the background penalty. For other parameters, we sequentially search for their values based on a training dataset and leave-one-out cross-validation. For each parameter, we do a grid search at 5 values (empirically selected).

## 4. Experiments

### 4.1. Datasets

To evaluate our method quantitatively, we build a new object instance segmentation dataset from the existing Polo dataset [3] for scene labeling. The dataset consists of 317 polo sport pictures, and originally has pixel-wise labeling for six object categories. We augment the original labeling by additional instance segmentation labels. In particular, we select the *horse* category, and manually generate all instance-level segmentation. The augmented dataset includes 842 object instances in the horse category, and the median of the number of instances per image is two.

### 4.2. Experiment setup

We follow the setting in [3] and select ten templates from the image subset that includes only one object instance.The selected templates cover six to seven typical viewpoints and

| Method | Mi-AP | Mi-AR | Ma-AP | Ma-AR | Avg-FP | Avg-FN |
|---|---|---|---|---|---|---|
| SL+CCA | 23.7 | 41.9 | 54.8 | 64.8 | 2.6 | 1.0 |
| 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. 'SL+CCA' 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.

a variety of poses. We initialize the object hypothesis set by Hough voting, which generates between 30 to 160 initial hypotheses. For each object instance, we build a Gaussian mixture appearance model with up to 15 components.

To evaluate the performance of our model, we build two baseline methods for comparison. The first baseline, denoted as 'SL+CCA', generates instance segmentation from the category-level labeling. We first predict the horse class label for all superpixels, and then run connected component analysis to group the class labels into instance segmentation. The second baseline is based on the initialization of our method. Instead of generating many hypotheses, we assume the number of objects in each image is known, and only keep that number of hypotheses with highest voting scores. Given the object center and mask information, we run the same GrabCut procedure to obtain the object instance segmentation. We start from the strongest hypothesis and greedily generate all instance labelings. This baseline is referred to as 'HV+GC'.

### 4.3. Segmentation performance

We evaluate against three different metrics on 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 we use a weak criterion as we care about segmentation: a false positive refers to an incorrect object detection (i.e., no overlapping pixels with the ground truth); and a miss detection refers to a ground-truth object that is completely missed.

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 variants of our model achieve significantly better micro average precision and recall than the two baselines. For macro average precision, the baseline 'HV+GC' has a higher score, which is not surprising since it knows the correct number of instances (but suffers in terms of macro average recall). We show some examples of our results on the Polo dataset in Figure 2.

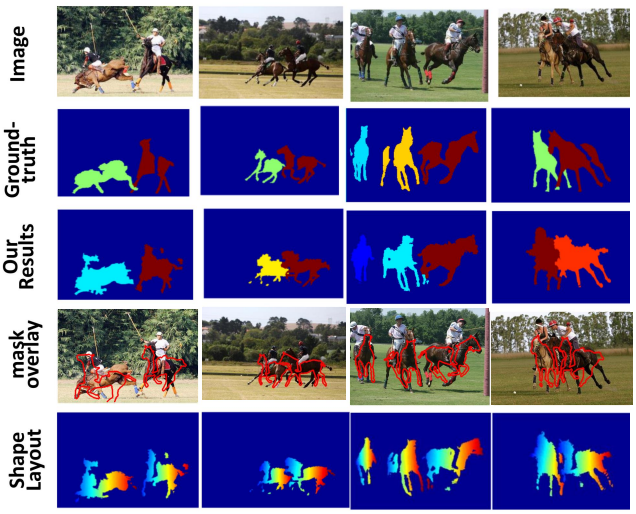


Figure 2. Examples of instance segmentation generated by our model on the Polo dataset. Note that color is only used to distinguish different objects.

## References


[1] O. Barinova, V. Lempitsky, and P. Kohli. On detection of multiple object instances using hough transforms. *PAMI*, 34(9):1773–1784, 2012.

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

[3] S. Gould and Y. Zhang. Patchmatchgraph: building a graph of dense patch correspondences for label transfer. In *ECCV*. 2012.

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

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

[6] C. Rother, V. Kolmogorov, and A. Blake. GrabCut: Interactive foreground extraction using iterated graph cuts. 2004.

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

[8] 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.

[9] J. Tighe and S. Lazebnik. SuperParsing: Scalable nonparametric image parsing with superpixels. 2010.

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

[11] 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.

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

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