Computer Vision and Image Understanding 114 (2010) 1055–1067

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Detecting object boundaries using low-, mid-, and high-level information

Songfeng Zheng [a], Alan Yuille [b], Zhuowen Tu [c,*]

[a] Department of Mathematics, Missouri State University 901 S. National Ave., Springfield, MO 65897, USA
[b] Department of Statistics, Department of Psychology, and Department of Computer Science, UCLA 8967 Math Sciences Bldg, Los Angeles, CA 90095, USA
[c] Lab of Neuro Imaging (LONI), Department of Neurology, and Department of Computer Science, UCLA 635 Charles E. Young Drive South, Los Angeles, CA 90095, USA

ARTICLE INFO

Article history:
Received 13 October 2008
Accepted 15 July 2010
Available online 13 August 2010

Keywords:
Boundary detection
Low-level information
High-level information
Shape matching
Cue integration

ABSTRACT

Object boundary detection is an important task in computer vision. Recent work suggests that this task can be achieved by combining low-, mid-, and high-level cues. But it is unclear how to combine them efficiently. In this paper, we present a learning-based approach which learns cues at different levels and combines them. This learning occurs in three stages. At the first stage, we learn low-level cues for object boundaries and regions. At the second stage, we learn mid-level cues by using the short and long range context of the low-level cues. Both these stages contain object-specific information – about the texture and local geometry of the object – but this information is implicit. In the third stage we use explicit high-level information about the object shape in order to further improve the quality of the object boundaries. The use of the high-level information also enables us to parse the object into different parts. We train and test our approach on two popular datasets – Weizmann horses [3] and ETHZ cows [24] – and obtain encouraging results. Although we have illustrated our approach on horses and cows, we emphasize that it can be directly applied to detect, segment, and parse other types of objects.

1. Introduction

Object boundary detection and foreground/background segmentation are important problems in computer vision, and they are often tightly coupled.

Local cues like gradients used in classical edge detectors (e.g. [4]) are often insufficient to characterize object boundaries [20,27]. For example, Fig. 1 shows the results of the Canny edge detector [4] applied to some natural images with cluttered backgrounds [3]. The edge map alone does not provide enough cues for segmenting the object. Marr [26] proposed a strategy for addressing this problem by combining low-, mid-, and high-level cues. However, despite some progress made in this direction [9,10,32,37], the problem remains unsolved.

Recent advances in machine learning had made it more practical to combine low-, mid-, and high-level cues for object detection. For example, Borenstein et al. [3] combined top-down information (learned configurations of image patches) with bottom-up approaches (intensity-based segmentation) in order to achieve foreground/background segmentation. In the image parsing framework [35], data-driven proposals (using low-level cues) were used to guide high-level generative models. Fergus et al. [14] built a top-down model based on features extracted by interest point operators. Conditional Markov random fields models [22,33] were used to enforce local consistency for labeling and object detection. Other approaches combine bottom-up and top-down learning in a loop [25]. OBJCUT [21] combined cues at different levels in order to perform object segmentation. He et al. [17] proposed a context-dependent conditional random field model to take context into account. In related work, Wang et al. [39,40] proposed a dynamic conditional random field model to incorporate context information for segmenting image sequences. More recently, Zhu et al. [41,42] built hierarchical models to incorporate semantic and context information at different levels.

These approaches have shown the effectiveness of of combining cues at different levels. But, when, where and how to combine cues from different levels is still unclear. For example, it is very difficult to build a generative appearance model, to capture the complex appearance patterns of the horses in Fig. 1; the patches used in [3,7,25] cannot deal with large scale deformations and they also have difficulties in capturing complex variations in appearance. Other approaches, like [16,17,21,23,39,40], lead to complex models which require solving time-consuming inference problems.

In this paper, we use a learning-based approach to learn and combine cues at different levels. This gives a straightforward method with a simple and efficient inference algorithm. More precisely, we use probabilistic boosting trees (PBTs) [34] (a variation of boosting [13]) for learning and combining low- and mid-level cues. Then we use a shape matching algorithm [36] to engage high-level shape information, and to parse the object into different components, e.g. head, back, legs, and other parts of horses or cows. Our strategy relates to Wolpert's work on stacking, which builds classifiers on top of other classifiers, but is very different in detail.

* Corresponding author.
E-mail address: ztu@loni.ucla.edu (Z. Tu).

1077-3142/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2010.07.004

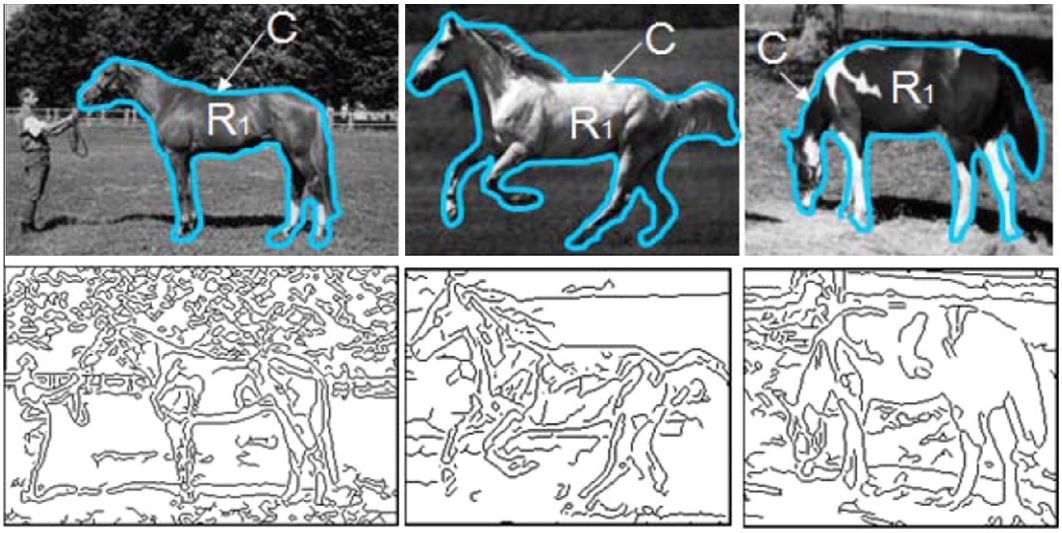

Fig. 1. Examples of the Weizmann horse dataset. The first row shows three typical images, each containing a horse, where C is the boundary we want to detect and R_1 denotes the foreground region. The second row displays edges detected by Canny edge detector at scale $\sigma = 1.0$.

We note that Ross and Kaelbling [29] also addresses segmentation using learning, but their approach is very different and involves motion cues and learning Markov random field models.

We compare our system with other approaches for this problem. The most directly comparable one is the work by Ren et al. [28] which gives detailed performance evaluations for combining low-, mid-, and high-level information. Our results show large improvement over their approach in many respects, particularly at the low- and mid-levels. It is less easy to make direct comparison with other works [3,7,21,25] because some of them [21] were not evaluated on large testing datasets, and the details of performance evaluation were not all given. Also some approaches [21,25] used color images. In [7], the authors first get a shortlist containing 10 candidates, and then pick the best one by hand, while our approach outputs only one result for each image. Hierarchical methods [41,42] obtain very good results but use more complex object models and require heavy inference.

2. Problem formulation

Given an image $\mathbf{I}$, we assume there is an object of interest in the foreground. The goal is to automatically detect the boundary of this object, and thus, perform foreground/background segmentation. In addition, it is desired to parse the object and identify its parts (e.g. head, leg, back, etc. of a horse or cow).

More precisely, we seek to decompose an image defined on a 2D image lattice Λ into two disjoint connected regions R_0, R_1 so that $R_0 \cup R_1 = \Lambda$ and $R_0 \cap R_1 = \emptyset$. R_0 is the background region and R_1 is the foreground (i.e. corresponding to the object). We denote a solution by:

$$W = (R_0, R_1), \quad R_0 \text{ background}, \quad R_1 \text{ foreground}. \tag{1}$$

We can also represent this by the object boundary curve $C = \partial R_1$ with the convention that the object is in the interior of the boundary, i.e. $R_1 = \text{interior}(C)$. In this paper, the object boundaries are closed curves and are represented by point sets.

2.1. The bayesian formulation

The optimal solution W^* for for this boundary detection task can be obtained by solving the Bayesian inference problem:

$$W^* = \arg\max_W p(W|\mathbf{I}) = \arg\max_W p(\mathbf{I}|R_0, R_1)p(R_0, R_1), \tag{2}$$

where $p(\mathbf{I}|R_0, R_1)$ models the image generating process in the foreground and background regions, and $p(R_0, R_1)$ defines the prior for the boundary contour. For example, we can use a probability model for the shape of the object.

However, it is difficult to use Eq. (2) directly because the image generating process is very complicated. Objects, such as horses and cows, have complex image appearance due to their varied texture patterns and the lighting conditions. Moreover, the background is even more varied and complex to model. Hence it is hard to model the image appearance $p(\mathbf{I}|R_0, R_1)$ directly although might be easier to model the boundary shape $p(R_0, R_1)$.

2.2. An alternative perspective

We avoid the difficulties above by defining the conditional distribution $P(W|\mathbf{I})$ directly:

$$P(W|\mathbf{I}) \propto \exp\{-E(W;\mathbf{I})\}.$$

Then we seek to estimate:

$$W^* = \arg\max P(W|\mathbf{I}) = \arg\min E(W;\mathbf{I}). \tag{3}$$

From the definition of C and W, finding the optimal W is equivalent to finding the optimal C. As such, we can rewrite Eq. (3) as

$$C^* = \arg\min E(C;\mathbf{I}),$$

where the energy function $E(C;\mathbf{I})$ is defined by:

$$E(C;\mathbf{I}) = E_{dis}(C;\mathbf{I}) + \tau E_{shape}(C), \tag{4}$$

where $E_{dis}(C;\mathbf{I})$ models the image appearance cues discriminatively, and $E_{shape}(C)$ models the boundary shape.

In our approach, the low- and mid-level cues are captured *implicity* by $E_{dis}(C;\mathbf{I})$. The high-level cues are represented *explicitly* by $E_{shape}(C)$, which is analogous to $-\log P(R_0, R_1)$ in the Bayesian formulation given by Eq. (2). The parameter τ balances the importance of $E_{dis}(C;\mathbf{I})$ and $E_{shape}(C)$ and is determined by cross-validation.

We define $E_{dis}(C;\mathbf{I})$ to be:

$$E_{dis}(C;\mathbf{I}) = -\sum_{\mathbf{r}\in\Lambda/C} \log p(\mathbf{I}(\mathbf{r}), y(\mathbf{r}) = 0|\mathbf{I}(\mathcal{N}(\mathbf{r})/\mathbf{r})) - \sum_{\mathbf{r}\in C} \log p(\mathbf{I}(\mathbf{r}), y(\mathbf{r}) = 1|\mathbf{I}(\mathcal{N}(\mathbf{r})/\mathbf{r}))), \tag{5}$$