Pattern Recognition 47 (2014) 1523–1534

Contents lists available at ScienceDirect

Pattern Recognition

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

Learning discriminative localization from weakly labeled data

Minh Hoai [a,*], Lorenzo Torresani [b], Fernando De la Torre [a], Carsten Rother [c]

[a] *Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA*
[b] *Department of Computer Science, Dartmouth College, Hanover, NH 03755 USA*
[c] *TU Dresden, Germany*

ARTICLE INFO

Article history:
Received 22 January 2013
Received in revised form
14 May 2013
Accepted 24 September 2013
Available online 4 October 2013

Keywords:
Discriminative discovery
Object detection
Event detection
Image classification
Time series classification
Weakly supervised learning

ABSTRACT

Visual categorization problems, such as object classification or action recognition, are increasingly often approached using a detection strategy: a classifier function is first applied to candidate subwindows of the image or the video, and then the maximum classifier score is used for class decision. Traditionally, the subwindow classifiers are trained on a large collection of examples manually annotated with masks or bounding boxes. The reliance on time-consuming human labeling effectively limits the application of these methods to problems involving very few categories. Furthermore, the human selection of the masks introduces arbitrary biases (e.g., in terms of window size and location) which may be suboptimal for classification. We propose a novel method for learning a discriminative subwindow classifier from examples annotated with binary labels indicating the presence of an object or action of interest, but *not* its location. During training, our approach simultaneously localizes the instances of the positive class and learns a subwindow SVM to recognize them. We extend our method to classification of time series by presenting an algorithm that localizes the most discriminative set of temporal segments in the signal. We evaluate our approach on several datasets for object and action recognition and show that it achieves results similar and in many cases superior to those obtained with full supervision.

1. Introduction

Object categorization systems aim at recognizing the classes of the objects present in an image, independently of the background. Early computer vision methods for object categorization attempted to build robustness to background clutter by using image segmentation as preprocessing. It was hoped that segmentation methods could partition images into their high-level constituent parts, and categorization could then be simply carried out as recognition of the object classes corresponding to the segments. This naive strategy to categorization floundered on the challenges presented by bottom-up image segmentation. The difficulty of partitioning an image into objects purely based on low-level cues is now well understood and it has led in recent years to a flourishing of methods where bottom-up segmentation is assisted by concurrent top-down recognition [1–6]. However, the application of these methods has been limited in practice by (a) the challenges posed by the acquisition of detailed ground truth segmentations needed to train these systems, and (b) the high computational complexity of semantic segmentation, which requires solving the classification problem at the pixel-level. An efficient alternative is provided by object detection methods, which can perform object localization without requiring pixel-level segmentation. Object detection algorithms operate by evaluating a classifier function at many different subwindows of the image and then predicting the object presence in subwindows with high-score. This methodology has been applied with great success to a wide variety of object classes [7–11]. Recent work [12] has shown that efficient computation of classification maxima over all possible subwindows of an image is even possible for highly sophisticated classifiers, such as Support Vector Machines (SVMs) with spatial pyramid kernels. Although great advances have been made in terms of reducing the computational complexity of object detection algorithms, their accuracy has remained dependent on the amount of human-annotated data available to train them. Subwindows (or bounding boxes) are obviously less-time consuming to collect than detailed segmentations. However, the dependence on human work for training inevitably limits the scalability of these methods. Furthermore, not only the amount of ground truth data but also the characteristics of the human selections may affect the detection. For example, it has been shown [8] that the specific size and location of the selections may have a significant impact on performance. In some cases, including a margin around the bounding box of the training selections will lead to better detection because of statistical

[*] Corresponding author. Current address: Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, UK. Tel.: +44 01865 283 057.
E-mail address: minhhoai@robots.ox.ac.uk (M. Hoai).

0031-3203/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.09.028

correlation between the appearance of the region surrounding the object (often referred to as the "spatial context") and the category of the object (e.g., cars tend to appear on roads). However, it is rather difficult to tune the amount ofcontext to include for optimal classification. The problem is even more acute for the case of categorization of time series. Consider the task of automatically monitoring the behavior of an animal based on its body movement. It is safe to believe that the intrinsic differences between the distinct animal activities (e.g., drinking, exploring) do not appear continuously in the examples but are rather associated with specific movement patterns (e.g., the turning of the head, a short fast-pace walk) possibly occurring multiple times in the sequences. Thus, as for the case of object categorization, classification based on comparisons of the whole signals is unlikely to yield good performance. However, if we asked a person to localize the most discriminative patterns in such sequences, we would obtain highly subjective annotations, unlikely to be optimal for the training of a classifier.

In this paper we propose a novel framework, based on multiple-instance learning [13,14], that simultaneously localizes the most discriminative subwindows in the data and learns a classifier to distinguish them. Our algorithm requires only the class labels as annotation for the training examples, and thus eliminates the high cost and arbitrariness of human ground truth selections. In the case of object categorization, our method optimizes an SVM classification objective with respect to both the classifier parameters and the subwindows containing the object of interest in the positive image examples. In the case of classification of time series, we relax the subwindow contiguity constraint in order to discover discriminative patterns which may occur discontinuously over the observation period. Specifically, we allow the discriminative patterns to occur in at most k disjoint time-intervals, where k is a problem-dependent tunable parameter of our system. The algorithm solves for the locations and durations of these intervals while learning the SVM classifier. We demonstrate our approach on several object and activity recognition datasets and show that our weakly supervised classifiers consistently match and often surpass the accuracy of SVMs trained under full supervision.

2. Previous work

This section reviews related work on weakly supervised localization and multiple instance learning.

2.1. Weakly supervised localization

Most prior work on weakly supervised object localization and classification is based on the use of region or part-based generative models. Fergus et al. [15] represent objects as flexible constellation of parts by learning probabilistic models of both the appearance as well as the mutual position of the parts. Parts are selected from points found by a feature detector. Classification of a test image is performed in a Bayesian fashion by evaluating the detected features using the learned model. The performance of this system rests completely on the ability of the feature detector to fire consistently at points corresponding to the learned parts of the model. Russell et al. [16] instead propose an unsupervised algorithm to discover objects and associated segments from a large collection of images. Multiple segmentations are computed from each image by varying the parameters of a segmentation method. The key-assumption is that each object instance is correctly segmented at least once and that the features of correct segments form object-specific coherent clusters discoverable using latent topic models from text analysis. Although the algorithm is shown to be able to discover many different types of objects, its effectiveness as a categorization technique is unclear. Another line of research on unsupervised segmentation is the so-called co-segmentation task [17], where the goal is to extract automatically a common region of interest from a pair of (or multiple) images, where the region of interest is a pixel-accurate segmentation. While recent work has shown quite good results, e.g., [18,19], the utilized objective functions were mostly hand-crafted, and furthermore these approaches have not been applied to object and time series categorization. Cao and Fei-Fei [20] further extend the latent topic model by assuming that a single topic model is responsible for generating the image patches within each region of the image, thus enforcing spatial coherence within each segment. Todorovic and Ahuja [21] describe a system that learns tree-based representations of multiscale image segmentations via a subtree matching algorithm. A multitude of algorithms based on Multiple Instance Learning (MIL) have been recently proposed for training object classifiers with weakly supervised data (see [13,14,22–26] for a sampling of these techniques). Most of these methods view images as bags of segments, traditionally computed using bottom-up segmentation or fixed partitioning of the image into blocks. Then MIL trains a discriminative binary classifier predicting the class of segments, under the assumption that each positive training image contains at least one true-positive segment (corresponding to the object of interest), while negative training images contain none. However, these approaches incur the same problem faced by the early segmentation-based recognition systems: segmentation from low-level cues is often unable to provide semantically correct segments. Galleguillos et al. [27] attempt to circumvent this problem by providing multiple segmentations to the MIL learning algorithm in the hope that one of them is correct. The approach we propose does not rely on unreliable segmentation methods as preprocessing. Instead, it performs localization while training the classifier. This approach has also been adopted in a number of recent works [28–30], proposed at the same time or after our initial work was published [31]. However, these methods require either more annotation (e.g., 10% of training images is fully annotated [30]) or stronger starting points (e.g., object detectors of other classes [29]), and they use different classifiers such as boosting [28], Conditional Random Fields [29], Structure-Output SVM [30]. Our work can also be viewed as an extension of feature selection methods, in which different features are selected for each example. The idea of *joint* feature selection and classifier optimization has been proposed before, but always in combination with strongly labeled data. Schweitzer [32] proposes a linear time algorithm to select jointly a subset of pixels and a set of eigenvectors that minimize the Rayleigh quotient in Linear Discriminant Analysis. Nguyen and De la Torre [33] propose a convex formulation to simultaneously select the most discriminative pixels and optimize the SVM parameters. However, both aforementioned methods require the training data to be well aligned and the same set of pixels is selected for every image. Felzenszwalb et al. [34] describe Latent SVM, a powerful classification framework based on a deformable part model. However, also this method requires knowing the bounding boxes of foreground objects during training. Finally, Blaschko and Lampert [35] use *supervised* structured learning to improve the localization accuracy of SVMs.

The literature on weakly supervised or unsupervised localization and categorization applied to time series is fairly limited compared to the object recognition case. Buehler et al. [36] learn British sign language using weakly aligned scripts. Zhong et al. [37] detect unusual activities in videos by clustering equal-length segments extracted from the video. The segments falling in isolated clusters are classified as abnormal activities. Fanti et al. [38] describe a system for unsupervised human motion recognition from videos.

Appearance and motion cues derived from feature tracking are used to learn graphical models of actions based on triangulated graphs. Niebles et al. [39] tackle the same problem but represent each video as a bag of video words, i.e., quantized descriptors computed at spatial-temporal interest points. An EM algorithm for topic models is then applied to discover the latent topics corresponding to the distinct actions in the dataset. Localization is obtained by computing the MAP topic of each word.

2.2. Multiple instance SVMs

This section reviews Multiple-Instance SVMs (MI-SVMs) [14], a particular type of multiple instance learning [13,22] which our method is based on. MI-SVMs input a set of positive bags $\{\mathcal{B}_i^+ | i=1,\dots,n^+\}$ and a set of negative bags $\{\mathcal{B}_j^- | j=1,\dots,n^-\}$ (see footnote 1 for an explanation of the notation).[1] Each positive bag contains at least one positive instance while no negative bag contains positive instances. MI-SVMs learn an SVM for classification by solving the following constraint optimization:

$$\underset{\mathbf{w},b}{\text{minimize}} \ \tfrac{1}{2}\|\mathbf{w}\|^2, \tag{1}$$

$$\text{s.t.} \quad \max_{\mathbf{x}\in\mathcal{B}_i^+} \mathbf{w}^T\varphi(\mathbf{x})+b \ge 1 \ \ \forall i=1,\dots,n^+, \tag{2}$$

$$\max_{\mathbf{x}\in\mathcal{B}_j^-} \mathbf{w}^T\varphi(\mathbf{x})+b \le -1 \ \ \forall j=1,\dots,n^-. \tag{3}$$

here $\varphi(\mathbf{x})$ denotes the feature vector for the instance $\mathbf{x}$. The constraints appearing in this objective state that each positive bag must contain at least one instance classified as positive, and that *all* instances in each negative bag must be classified as negative. The goal is then to maximize the margin subject to these constraints. By optimizing this problem MI-SVMs obtain an SVM, i.e., parameters $(\mathbf{w},b)$. As in the traditional formulation of SVM, the constraints are allowed to be violated by introducing slack variables:

$$\underset{\mathbf{w},b,\{\alpha_i\},\{\beta_j\}}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n^+}\alpha_i + C\sum_{j=1}^{n^-}\beta_j, \tag{4}$$

$$\text{s.t.} \quad \max_{\mathbf{x}\in\mathcal{B}_i^+} \mathbf{w}^T\varphi(\mathbf{x})+b \ge 1-\alpha_i \ \ \forall i=1,\dots,n^+, \tag{5}$$

$$\max_{\mathbf{x}\in\mathcal{B}_j^-} \mathbf{w}^T\varphi(\mathbf{x})+b \le -1+\beta_j \ \ \forall j=1,\dots,n^-, \tag{6}$$

$$\alpha_i \ge 0 \ \ \forall i=1,\dots,n^+,$$
$$\beta_j \ge 0 \ \ \forall j=1,\dots,n^-.$$

here C is the parameter controlling the trade-off between having a large margin and less constraint violation.

3. Localization–classification SVM

In this section we first propose an algorithm to simultaneously localize objects of interest and train an SVM. We then extend it to classification of time series by presenting an efficient algorithm to identify in the signal an optimal set of discriminative segments, which are not constrained to be contiguous.

3.1. The learning objective

Assume we are given a set of positive training images $\{\mathbf{d}_i^+ | i=1,\dots,n^+\}$ and a set of negative training images $\{\mathbf{d}_j^- | j=1,\dots,n^-\}$ corresponding to weakly labeled data with labels indicating for each example the presence or absence of an object of interest. Let $\mathcal{LS}(\mathbf{d})$ denote the set of all possible subwindows of image $\mathbf{d}$. For a subwindow $\mathbf{x}\in\mathcal{LS}(\mathbf{d})$, let $\varphi(\mathbf{x})$ be the feature vector computed from the image subwindow.

We use MI-SVM to learn an SVM for joint localization and classification by setting $\mathcal{B}_i^+ = \mathcal{LS}(\mathbf{d}_i^+)$, $\mathcal{B}_j^- = \mathcal{LS}(\mathbf{d}_j^-)$. This reflects the requirement that each positive image must contain at least one subwindow classified as positive, and that *all* subwindows in each negative image must be classified as negative. The goal is then to maximize the margin subject to these constraints. By optimizing this problem we obtain an SVM, i.e., parameters $(\mathbf{w},b)$, that can be used for localization and classification. Given a new testing image $\mathbf{d}$, localization and classification are done as follows. First, we find the subwindow $\hat{\mathbf{x}}$ yielding the maximum SVM score:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}\in\mathcal{LS}(\mathbf{d})}{\text{argmax}} \ \mathbf{w}^T\varphi(\mathbf{x}). \tag{7}$$

If the value of $\mathbf{w}^T\varphi(\hat{\mathbf{x}})+b$ is positive, we report $\hat{\mathbf{x}}$ as the detected object for the test image. Otherwise, we report no detection.

Our objective is in general non-convex. We optimize this objective using a coordinate descent approach that alternates between optimizing the objective w.r.t. parameters $(\mathbf{w}, b, \{\alpha_i\}, \{\beta_j\})$ and finding the subwindows of positive images $\{\mathbf{d}_i^+\}$ that maximize the SVM scores. This alternating approach is guaranteed to converge to a critical point. Every iteration of this alternating approach requires optimizing the objective w.r.t. parameters $\mathbf{w}, b, \{\alpha_i\}, \{\beta_j\}$ while fixing the subwindows of images $\{\mathbf{d}_i^+\}$. This sub-problem is convex, but the cardinality of the sets of all possible subwindows of negative images may be very large. Therefore, special treatment is required for constraints (6). We use *constraint generation* (i.e., the cutting plane algorithm) to handle these constraints [40]: $\mathcal{LS}(\mathbf{d}_j^-)$ is iteratively updated by adding the most violated constraint at every step. Although constraint generation has exponential running time in the worst case, it often works well in practice. This optimization algorithm was also proposed by Yu and Joachims [41], at the same time that this work was developed [31]. We refer the reader to [41] for a detailed description of the algorithm. A simple initialization approach is to use the entire positive images as starting subwindows. This works sufficiently well for the experiments described in Section 4, but better initialization approaches can be used (e.g., [42]).

Our optimization algorithm requires at each iteration to localize the subwindow maximizing the SVM score in each image. Thus, we need a very fast localization procedure. For this purpose, we adopt the representation and algorithm described in [12]. Images are represented as bags of visual words obtained by quantizing SIFT descriptors [43] computed at random locations and scales. For quantization, we use a visual dictionary built by applying K-means clustering to a set of descriptors extracted from the training images [44]. The set of possible subwindows for an image is taken to be the set of axis-aligned rectangles. The feature vector $\varphi(\mathbf{x})$ is the histogram of visual words associated with descriptors inside rectangle $\mathbf{x}$. Lampert et al. [12] showed that, when using this image representation, the search for the rectangle maximizing the SVM score can be executed efficiently by means of a branch-and-bound algorithm.

3.2. Extension to time series

As in the case of image categorization, even for time series the global statistics computed from the entire signal may yield

[1] Bold lowercase letters denote a column vector (e.g., $\mathbf{d}, \boldsymbol{\alpha}$). d_i, α_i represent the ith entries of the column vectors $\mathbf{d}$ and $\boldsymbol{\alpha}$, respectively. Non-bold letters represent scalar variables (e.g., C, α_i).

suboptimal classification. For example, the differences between two classes of temporal signals may not be visible over the entire observation period. However, unlike in the case of images where objects often appear as fully connected regions, the patterns of interest in temporal signals may not be contiguous. This raises a technical challenge when extending the learning formulation of (4) to time series classification: how to efficiently search for sets of non-contiguous discriminative segments? In this section we describe a representation of temporal signals and a novel efficient algorithm to address this challenge.

3.2.1. *Representation of time series*

Time series can be represented by descriptors computed at spatial-temporal interest points [45,46,39]. As in the case of images, sample descriptors from training data can be clustered to create a visual-temporal vocabulary [46]. Subsequently, each descriptor is represented by the ID of the corresponding vocabulary entry and the frame number at which the point is detected. In this work, we define a *k-segmentation* of a time series as a set of k disjoint time-intervals, where k is a tunable parameter of the algorithm. Note that it is possible for some intervals of a k-segmentation to be empty. Given a k-segmentation $\mathbf{x}$, let $\varphi(\mathbf{x})$ denote the histogram of visual-temporal words associated with interest points in $\mathbf{x}$. Let C_i denote the set of words occurring at frame i. Let $a_i = \sum_{c \in C_i} w_c$ if C_i is non-empty, and $a_i = 0$ otherwise. a_i is the weighted sum of words occurring in frame i where word c is weighted by SVM weight w_c. From these definitions it follows that $\mathbf{w}^T \varphi(\mathbf{x}) = \sum_{i \in \mathbf{x}} a_i$. For fast localization of discriminative patterns in time series we need an algorithm to efficiently find the k-segmentation maximizing the SVM score $\mathbf{w}^T \varphi(\mathbf{x})$. Indeed, this optimization can be solved globally in a very efficient way. The following section describes the algorithm.

3.2.2. *An efficient localization algorithm*

Let n be the length of the time signal and $\mathcal{I} = \{[l,u] : 1 \le l \le u \le n\}$ be the set of all subintervals of $[1,n]$. For a subset $S \subseteq \{1,\dots,n\}$, let $f(S) = \sum_{i \in S} a_i$. Maximization of $\mathbf{w}^T \varphi(\mathbf{x})$ is equivalent to

$$\underset{I_1,\dots,I_k}{\text{maximize}} \sum_{j=1}^{k} f(I_j) \quad \text{s.t. } \mathrm{I}_i \in \mathcal{I} \ \& \ \mathrm{I}_i \cap \mathrm{I}_j = \emptyset \quad \forall \mathrm{i} \neq \mathrm{j}. \tag{8}$$

This problem can be optimized very efficiently using Algorithm 1 presented below.

Algorithm 1. Find best k disjoint intervals that optimize (8).

```
Input: $a_1, \dots, a_n$, $k \ge 1$.
Output: a set $\mathcal{X}^k$ of best $k$ disjoint intervals.
1: $\mathcal{X}^0 := \emptyset$.
2: for $m = 0$ to $k-1$ do
3:    $J_1 := \arg\max_{J \in \mathcal{I}} f(J)$ s.t. $J \cap S = \emptyset \ \forall S \in \mathcal{X}^m$.
4:    $J_2 := \arg\max_{J \in \mathcal{I}} -f(J)$ s.t. $J \subset S \in \mathcal{X}^m$.
5:    if $f(J_1) \ge -f(J_2)$ then
6:       $\mathcal{X}^{m+1} := \mathcal{X}^m \cup \{J_1\}$
7:    else
8:       Let $S \in \mathcal{X}^m : J_2 \subset S$. $S$ is divided into three disjoint
         intervals: $S = S^- \cup J_2 \cup S^+$.
9:       $\mathcal{X}^{m+1} := (\mathcal{X}^m - \{S\}) \cup \{S^-, S^+\}$
10:   end if
11: end for
```

This algorithm progressively finds the set of m intervals (possibly empty) that maximize (8) for $m = 1, \dots, k$. Given the optimal set of m intervals, the optimal set of $m+1$ intervals is obtained as follows. First, find the interval J_1 that has maximum score $f(J_1)$ among the intervals that do not overlap with any currently selected interval (line 3). Second, locate J_2, the worst subinterval of all currently selected intervals, i.e., the subinterval with lowest score $f(J_2)$ (line 4). Finally, the optimal set of $m+1$ intervals is constructed by executing either of the following two operations, depending on which one leads to the higher objective:

1. Add J_1 to the optimal set of m intervals (line 6).
2. Break the interval of which J_2 is a subinterval into three intervals and remove J_2 (line 9).

Algorithm 1 assumes J_1 and J_2 can be found efficiently. This is indeed the case. We now describe the procedure for finding J_1. The procedure for finding J_2 is similar.

Let $\overline{\mathcal{X}^m}$ denote the relative complement of $\mathcal{X}^m$ in $[1,n]$, i.e., $\overline{\mathcal{X}^m}$ is the set of intervals such that the "union" of the intervals in $\mathcal{X}^m$ and $\overline{\mathcal{X}^m}$ is the interval $[1,n]$. Since $\mathcal{X}^m$ has at most m elements, $\overline{\mathcal{X}^m}$ has at most $m+1$ elements. Since J_1 does not intersect with any interval in $\mathcal{X}^m$, it must be a subinterval of an interval of $\overline{\mathcal{X}^m}$. Thus, we can find J_1 as $J_1 = \arg\max_{S \in \overline{\mathcal{X}^m}} f(J^S)$ where

$$J^S = \arg\max_{J \subseteq S} f(J). \tag{9}$$

Eq. (9) is a basic operation that is needed to be performed repeatedly: finding a subinterval of an interval that maximizes the sum of elements in that subinterval. This operation can be performed by Algorithm 2 below with running time complexity $\mathbf{O}(n)$.

Algorithm 2. Find the best subinterval.

```
Input: $a_1, \dots, a_n$, an interval $[l,u] \subset [1,n]$.
Output: $[sl, su] \subset [l,u]$ with maximum sum of elements.
b_0 := 0.
for $m = 1$ to $n$ do
   $b_m := b_{m-1} + a_m$. //compute integral image
end for
$[sl, su] := [0,0]$; $val := 0$. //empty subinterval
$\widehat{m} := l - 1$. //index for minimum element so far
for $m = l$ to $u$ do
   if $b_m - b_{\widehat{m}} > val$ then
      $[sl, su] := [\widehat{m}+1; m]$; $val := b_m - b_{\widehat{m}}$
   else if $b_m < b_{\widehat{m}}$
      $\widehat{m} := m$. //keep track of the minimum element
   end if
end for
```

Note that the result of executing (9) can be cached; we do not need to recompute J^S for many S at each iteration of Algorithm 1. Thus the total running complexity of Algorithm 1 is $\mathbf{O}(nk)$. Algorithm 1 guarantees to produce a globally optimal solution for (8), as proved in the following section.

3.2.3. *Global optimality of Algorithm 1*

Algorithm 1 guarantees to produce a globally optimal solution for (8). Even stronger, the set $\mathcal{X}^m = \{I_1^m, \dots, I_m^m\}$ produced by the algorithm is the set of best m intervals that maximize (8). This section sketches a proof by induction. A reader who is not interested in the proof can skip this section.

$(+)$ $m = 1$, this can be easily verified.

$(+)$ Suppose $\mathcal{X}^m$ is the set of best m intervals that maximize (8). We now prove that $\mathcal{X}^{m+1}$ is optimal for $m+1$ intervals. Assume the contrary, $\mathcal{X}^{m+1}$ is not optimal for $m+1$ intervals. There exist disjoint

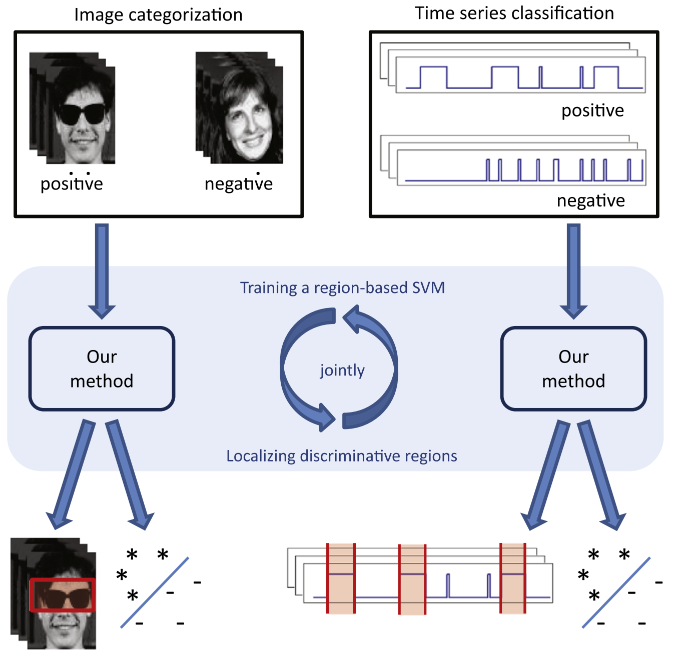
Image categorization
positive
negative
Time series classification
positive
negative
Training a region-based SVM
Our method
jointly
Our method
Localizing discriminative regions

Fig. 1. A unified framework for image categorization and time series classification from weakly labeled data. Our method simultaneously localizes the regions of interest in the examples and learns a region-based classifier, thus building robustness to background and uninformative signal.

Fig. 2. Examples taken from (a) the CMU Face Images and (b) the street scene dataset.

intervals $T_1, \ldots, T_{m+1}$ such that

$$\sum_{i=1}^{m+1} f(T_i) > \sum_{i=1}^{m+1} f(I_i^{m+1}). \tag{10}$$

Because the way we construct $\mathcal{X}^{m+1}$ from $\mathcal{X}^m$, we have

$$\sum_{i=1}^{m+1} f(I_i^{m+1}) = \sum_{i=1}^{m} f(I_i^m) + \max\{f(J_1), -f(J_2)\},$$

$$\text{where } J_1 = \arg\max_{J \in \mathcal{I}} f(J) \quad \text{s.t. } J \cap I_i^m = \emptyset \ \forall i, \tag{11}$$

$$J_2 = \arg\max_{J \in \mathcal{I}} -f(J) \quad \text{s.t. } J \subset I_i^m \text{ for an } i. \tag{12}$$

This, together with (10), leads to

$$\max\{f(J_1), -f(J_2)\} < \sum_{i=1}^{m+1} f(T_i) - \sum_{i=1}^{m} f(I_i^m). \tag{13}$$

Consider the overlapping between $T_1, \ldots, T_{m+1}$ and $I_1^m, \ldots, I_m^m$, there are two cases.

- *Case*1: $\exists j : T_j \cap I_i^m = \emptyset \ \forall i$. In this case, we have

$$f(T_j) \le f(J_1) < \sum_{i=1}^{m+1} f(T_i) - \sum_{i=1}^{m} f(I_i^m), \tag{14}$$

Table 1
Comparison results on the CMU Face and car datasets. BoW: bag of words approach [50]. SVM: SVM using global statistics. SVM-FS [12] requires bounding boxes of foreground objects during training. Our method is significantly better than the others, and it outperforms even the algorithm using strongly labeled data.

Dataset	Measure	BoW	SVM	SVM-FS	Ours
Faces	Acc. (%)	80.11	82.97	86.79	**90.0**
	ROC area	n/a	0.90	0.94	**0.96**
Cars	Acc. (%)	77.5	80.75	81.44	**84.0**
	ROC area	n/a	0.86	0.88	**0.90**

$$\Rightarrow \sum_{i=1}^{m} f(I_i^m) < \sum_{i=1,m+1,i \neq j} f(T_i). \tag{15}$$

This contradicts with the assumption that $\{\mathbf{I}_1^m, \ldots, \mathbf{I}_m^m\}$ is the set of best m intervals that maximize (8).

- *Case*2: $\forall j, \exists i : T_j \cap I_i^m \neq \emptyset$. Since there are $m+1$ T_j's, and there are only m I_i^m's, there must exist one i s.t. I_i^m intersects with at least two of T_j's. Suppose l, l_1, l_2 are indices s.t. $T_{l_1} \cap I_l^m \neq \emptyset$ and $T_{l_2} \cap I_l^m \neq \emptyset$. Furthermore, suppose T_{l_1}, T_{l_2} are consecutive intervals of T_j's (T_{l_1} precedes T_{l_2} and there is no T_j in between). Let $T_{l_1} = [t_{l_1}^-, t_{l_1}^+]$, $T_{l_2} = [t_{l_2}^-, t_{l_2}^+]$. Consider the interval $T = [t_{l_1}^+ + 1, t_{l_2}^- - 1]$. Because $T_{l_1} \cap I_l^m \neq \emptyset$ and $T_{l_2} \cap I_l^m \neq \emptyset$, T must be a subinterval of I_l^m, i.e., $T \subset I_l^m$. Hence

$$-f(T) \le -f(J_2) < \sum_{i=1}^{m+1} f(T_i) - \sum_{i=1}^{m} f(I_i^m), \tag{16}$$

$$\Rightarrow \sum_{i=1}^{m} f(I_i^m) < f(T) + \sum_{i=1}^{m+1} f(T_i), \tag{17}$$

$$\Rightarrow \sum_{i=1}^{m} f(I_i^m) < f(\underbrace{T_{l_1} \cup T \cup T_{l_2}}_{\text{an interval}}) + \sum_{i \neq l_1, l_2} f(T_i). \tag{18}$$

This contradicts with the assumption that $\{\mathbf{I}_1^m, \ldots, \mathbf{I}_m^m\}$ is the best set of m intervals that maximize (8).

Since both cases lead to a contradiction, $\mathcal{X}^{m+1}$ must be the best set of $m+1$ intervals that maximize (8). This completes the proof.

3.3. Multi-class categorization

The formulation presented in Section 3.1 can be extended to handle multiple classes, by replacing binary SVMs with multi-class SVMs [47]. Previous work for multi-class multiple instance learning exists [48,49], but has not been used for discriminative localization.

Assume we are given a set of training images (or time series) $\{\mathbf{d}_i | i = 1, \ldots, n\}$ with corresponding class labels $\{l_i | i = 1, \ldots, n\}$. The label $l_i \in \{1, \ldots, m\}$ indicates that the image $\mathbf{d}_i$ contains an object instance of category l_i. We learn an SVM for joint localization and classification by solving the following constrained optimization:

$$\underset{\{\mathbf{w}_j\},\{\xi_i\}}{\text{minimize}} \ \frac{1}{2m} \sum_{j=1}^{m} \|\mathbf{w}_j\|^2 + C \sum_{i=1}^{n} \xi_i \tag{19}$$

$$\text{s.t.} \quad \max_{\mathbf{x} \in \mathcal{LS}(\mathbf{d}_i)} \mathbf{w}_{l_i}^T \varphi(\mathbf{x}) \ge \max_{\mathbf{x} \in \mathcal{LS}(\mathbf{d}_i)} \mathbf{w}_j^T \varphi(\mathbf{x}) + 1 - \xi_i$$

$$\forall i \in \{1, \ldots, n\}, \forall j \in \{1, \ldots, m\} \backslash \{l_i\},$$

$$\xi_i \ge 0 \ \forall i \in \{1, \ldots, n\}.$$

The constraints appearing in this objective state that for each image $\mathbf{d}_i$, the detector of the correct class (l_i) should output a classification score higher than those produced by detectors of the other classes. Here, $\{\xi_i\}$ are slack variables, and C is the parameter controlling the trade-off between having a large margin and less constraint violation. The goal is then to

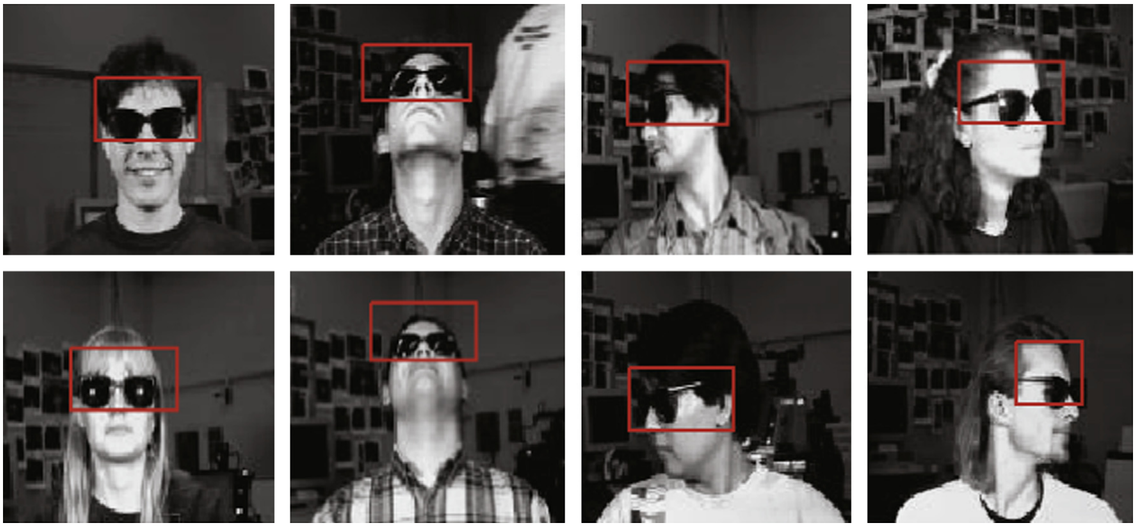

Fig. 3. Localization of sunglasses on test images.

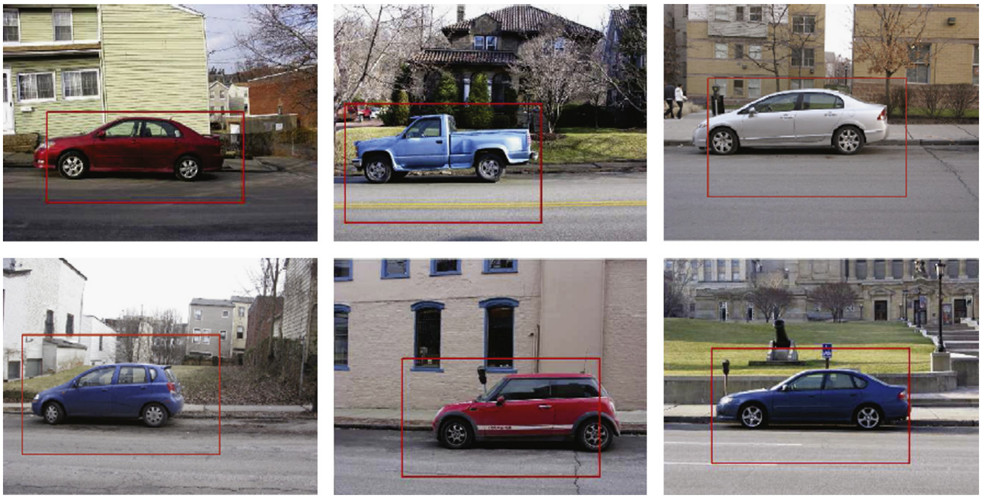

Fig. 4. Localization of cars on test images. Note how the road below the cars is partially included in the detection output. This indicates that the appearance of road serves as a contextual indication for the presence of cars.

maximize the margin subject to these constraints. By optimizing this problem we obtain a multi-class SVM, i.e., parameters $(\mathbf{w}_1, \dots, \mathbf{w}_m)$, that can be used for localization and categorization. Given a new testing image $\mathbf{d}$, localization and categorization are done as follows. First, we find the category $\hat{j}$ and subwindow $\hat{\mathbf{x}}$ yielding the maximum SVM score:

$$\hat{j}, \hat{\mathbf{x}} = \underset{j, \mathbf{x} \in \mathcal{LS}(\mathbf{d})}{\operatorname{argmax}} \mathbf{w}_j^T \varphi(\mathbf{x}). \tag{20}$$

We report $\hat{\mathbf{x}}$ as the detected object of category $\hat{j}$ for the test image.

4. Experiments

This section describes experiments on several datasets for object categorization and time series classification (Fig. 1).

4.1. Object localization and categorization

4.1.1. Experiments on car and face datasets

This subsection presents evaluations on two image collections. The first experiment was performed on CMU Face Images, a publicly available dataset from the UCI machine learning repository.[2] This database contains 624 face images of 20 people with different expressions and poses. The subjects wear sunglasses in roughly half of the images. Our classification task was to distinguish between the faces with sunglasses and the faces without sunglasses. Some image examples from the database are given in Fig. 2(a). We divided this image collection into disjoint training and testing subsets. Images of the first 8 people were used for training while images of the last 12 people were reserved for testing. Altogether, we had 254 training images (126 with glasses and 128 without glasses) and 370 testing images (185 examples for both the positive and the negative class).

The second experiment was performed on a dataset collected by us. Our collection contains 400 images of street scenes. Half of the images contain cars and half of them do not. This is a challenging dataset because the appearance of the cars in the images varies in shape, size, grayscale intensity, and location. Furthermore, the cars occupy only a small portion of the images and may be partially occluded by other objects.

[2] http://archive.ics.uci.edu/ml/datasets/CMU+Face+Images

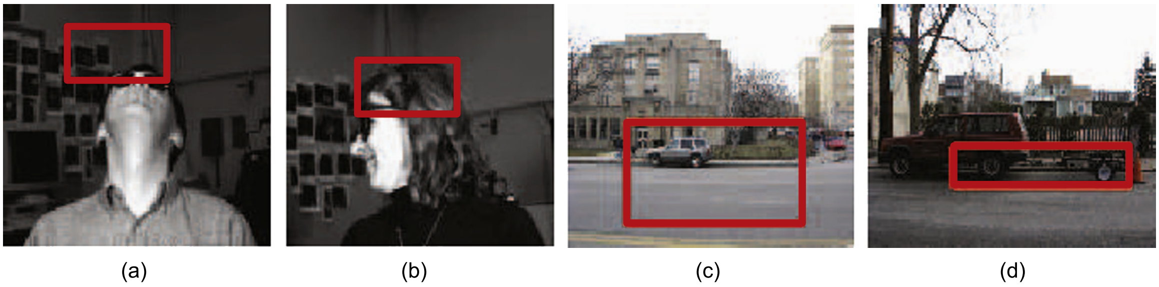

Fig. 5. Difficult cases for localization. (a, b) Sunglasses are not clearly visible in the images. (c) The foreground object is very small. (d) Misdetection due to the presence of the trailer wheel.

Table 2
Results of binary classification between each of the four classes of Caltech-4 and the background clutter class. BoW: bag of word approach [50]. SVM: traditional SVM using global statistics. SVM-FS [12] is the SVM method that requires strongly labeled data during training. SVM-FS++ is similar to SVM-FS, but the manually provided bounding boxes are extended to contain some background; the extended height and width is 1.5 the original height and width. Results of SVM-FS and SVM-FS++ for the Cars class is displayed as n/a because of the unavailability of ground truth annotation.

Class	Measure	BoW	SVM	SVM-FS	SVM-FS++	Ours
Airplanes	Acc. (%)	89.74	**96.05**	89.40	93.91	**96.05**
	ROC area	n/a	**0.99**	0.95	0.98	**0.99**
Cars	Acc. (%)	94.93	98.17	n/a	n/a	**98.28**
	ROC area	n/a	**1.00**	n/a	n/a	**1.00**
Faces	Acc. (%)	59.83	88.70	86.78	85.04	**89.57**
	ROC area	n/a	**0.95**	0.91	0.90	**0.95**
Motorbikes	Acc. (%)	76.80	**88.99**	84.67	84.80	87.81
	ROC area	n/a	**0.95**	0.92	0.91	0.94

Some examples of images from this dataset are shown in Fig. 2(b). Given the limited amount of examples available, we applied 4-fold cross validation to obtain an estimate of the performance.

Each image was represented by a set of 10,000 local SIFT descriptors [43] selected at random locations and scales. The descriptors were quantized using a dictionary of 1000 visual words obtained by applying hierarchical *K*-means [50] to 100,000 training descriptors.

In order to speed up the learning, we reduce the space of subwindows by imposing an upper constraint on the rectangle size. In the first experiment, as the image size is 120 × 128 and the sizes of sunglasses are relative small, we restricted the height and width of permissible rectangles to not exceed 30 and 50 pixels, respectively. Similarly, for the second experiment, we constrained permissible rectangles to have height and width no larger than 300 and 500 pixels, respectively (c.f. image size of 600 × 800). We also notice that these upper constraints prevent the optimization algorithm from making aggressive updates (which often lead to early termination at a local minimum). Notably, the imposing upper size is relatively large, compared with the sizes of the foreground objects.

We compared our approach to several competing methods. *SVM* denotes a traditional SVM approach in which each image is represented by the histogram of the words in the whole image. *BoW* is the *bag-of-words* method [51,44,50] in the implementation of [52]. It uses a 10-nearest neighbor classifier. We also benchmarked our method against *SVM-FS* [12], a fully supervised method requiring ground truth subwindows during training (FS stands for fully supervised). *SVM-FS* trains an SVM using ground truth bounding boxes as positive examples and ten random rectangles from each negative image for negative data.

Table 1 shows the classification performance measured using both the accuracy rates and the areas under the ROCs. Note that our approach outperforms not only *SVM* and *BoW* (which are based on global statistics), but also *SVM-FS*, which is a fully supervised method requiring the bounding boxes of the objects during training. This suggests that the boxes tightly enclosing the objects of interest are not always the most discriminative regions.

Our method automatically localizes the subwindows that are most discriminative for classification. Fig. 3 shows discriminative detection on a few face testing examples. Sunglasses are the distinguishing elements between positive and negative classes. Our algorithm successfully discovers such regions and exploits them to improve the classification performance. Fig. 4 shows some examples of car localization. Parts of the road below the cars tend to be included in the detection output. This suggests that the appearance of roads is a contextual indication for the presence of cars. Fig. 5 displays several difficult cases where our method does not provide good localization of the objects.

SVM , *SVM-FS*, and our proposed method require tuning of a single parameter, *C*, controlling the trade-off between a large margin and less constraint violation. This parameter was tuned using 4-fold cross validation on training data. The parameter sweeping was done exactly in the same fashion for all algorithms. Optimizing (4) was an iterative procedure, where each iteration involved solving a convex quadratic programming problem. Our implementation[3] used `CVX`, a package for specifying and solving convex programs [53,54]. We also used `Ilog Cplex`[4] for quadratic programming. We found that our algorithm generally converged within 100 iterations of coordinate descent.

4.1.2. *Experiments on Caltech-4*

This subsection describes an experiment on the publicly available[5] Caltech-4 dataset. This collection contains images of different categories: airplanes_side, cars_brad, faces, motorbikes_side, and background clutter. We consider binary classification tasks where the goal is to distinguish one of the four object classes (airplanes_side, cars_brad, faces, and motorbikes_side) from the background clutter class. In this experiment, we randomly sampled a set of 100 images from each class for training. The set of the remaining images was split into equal-size testing and validation sets. The validation data was used for parameter tuning.

Table 2 shows the results of this experiment. As shown, *SVM-FS*, a method that requires bounding boxes of the foreground objects for training, does not perform as well as *SVM* which is based on global statistics from the whole image. This result suggests that contextual information is very important for classification tasks on

[3] www.robots.ox.ac.uk/~minhhoai/downloads.html

[4] www-01.ibm.com/software/integration/optimization/cplex-optimizer/

[5] http://www.robots.ox.ac.uk/~vgg/data3.html

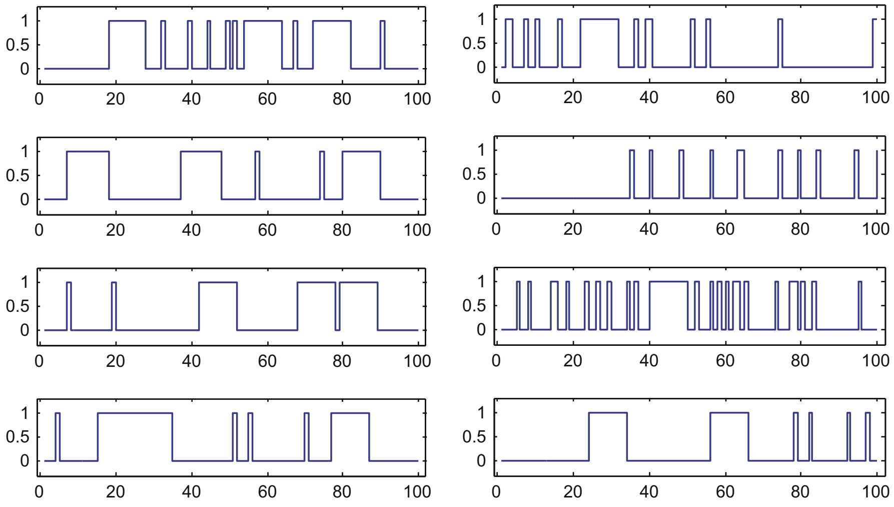

Fig. 6. What distinguishes the time series on the left from the ones on the right? Left: Positive examples, each containing three long segments with value 1 at random locations. Right: Negative examples, each containing fewer than three long segments with value 1. All signals are perturbed with measurement noise corresponding to spikes with value 1 at random locations.

Table 3
Classification performance on temporal data using our approach. We show the accuracy rates and the ROC areas obtained using different values of k, the number of discriminative time intervals used by the algorithm. Here traditional SVM, based on the global statistics of the signals, yields an accuracy rate of 66.5% and an area under the ROC of 0.577.

k	1	2	3–7	8	12	16	20
Acc.(%)	77.0	93.0	100	98.5	91.5	77.5	67.25
ROC area	0.843	0.980	1.00	0.998	0.933	0.793	0.613

this dataset. Indeed, it is easy to verify by visual inspection that the image backgrounds here often provide very strong categorization cues (see e.g., the almost constant background of the face images). As a result our method cannot provide any significant advantage on this dataset. However note that, unlike SVM-FS, our joint localization and classification does not harm the classification performance as our algorithm automatically learns the importance of contextual information and uses large subwindows for recognition. Having realized the importance of contextual information, we perform an additional experiment where the manually annotated object bounding boxes are uniformly extended to contain some background. This method is referred to as *SVM-FS*++ in Table 2, and it yields mixed results. It increases the performance of *SVM-FS* on Airplanes but decreases the performance or gains little improvement on the other datasets.

4.2. *Classification of time series*

This section describes our classification experiments on time series datasets.

4.2.1. *A synthetic example*

The data in this evaluation consists of 800 artificially generated examples of binary time series (400 positive and 400 negative). Some examples are shown in Fig. 6. Each positive example contains three long segments of fixed length with value 1. We refer to these as the foreground segments. Note that the end of a foreground segment may meet the beginning of another one, thus creating a longer foreground segment (see e.g., the bottom left signal of Fig. 6). The locations of the foreground segments are randomly distributed. Each negative example contains fewer than three foreground segments. Both positive and negative data are artificially degraded to simulate measurement noise: with a certain probability, zero energy values are flipped to have value 1. The temporal length of each signal is 100 and the length of each foreground segment is 10. We split the data into separate training and testing sets, each containing 400 examples (200 positive, 200 negative).

We evaluated the ability of our algorithm to discover automatically the discriminative segments in these weakly labeled examples. We trained our localization–classification SVM by learning k-segmentations for values of k ranging from 1 to 20. Note that the algorithm has no knowledge of the length or the type of the pattern distinguishing the two classes. Table 3 summarizes the performance of our approach. Traditional SVM, based on the statistics of the whole signals, yields an accuracy rate of 66.5% and an area under the ROC of 0.577. Thus, our approach provides much better accuracy than SVM. Note that the performance of our method is relatively insensitive to the choice of k, the number of discriminative time-intervals used for classification. It achieves 100% accuracy when the number of intervals are in the range 3–7; it works relatively well even for other settings. In practice, one can use cross validation to choose the appropriate number of segments. Furthermore, Table 3 reaffirms the need for using multiple intervals: our classifier built with only one interval achieves only an accuracy rate of 77%.

4.2.2. *Mouse behavior*

We now describe an experiment of mouse behavior recognition performed on a publicly available dataset.[6] This collection contains videos corresponding to five distinct mouse behaviors: drinking, eating, exploring, grooming, and sleeping. There are seven groups of videos, corresponding to seven distinct recording sessions. Because of the limited amount of data, performance is estimated using leave-one-group-out cross validation. This is the same evaluation methodology used by Dollár et al. [46]. Fig. 7 shows some representative frames of the clips. Refer to [46] for further details about this dataset.

We represented each video clip as a set of *cuboids* [46] which were spatial–temporal local descriptors. From each video we extracted cuboids at interest points computed using the cuboid

[6] http://vision.ucsd.edu/~pdollar/research/research.html

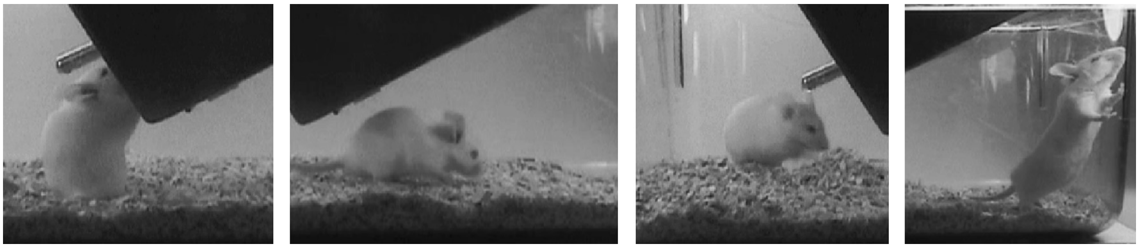

Fig. 7. Example frames from the mouse videos.

Table 4
F1 scores: detection performance of several algorithms. Higher F1 scores indicate better performance.

Action	[46]	1-NN	SVM	Ours
Drink	0.63	0.58	0.63	**0.67**
Eat	**0.92**	0.87	0.91	0.91
Explore	0.80	0.79	**0.85**	**0.85**
Groom	0.37	0.23	0.44	**0.54**
Sleep	0.88	0.95	**0.99**	**0.99**

detector [46]. To these descriptors we added cuboids computed at random locations in order to yield a total of 2500 points for each video (this augmentation of points was done to cancel out effects due to differing sequence lengths). A library of 50 cuboid prototypes was created by clustering cuboids sampled from training data using *K*-means. Subsequently, each cuboid was represented by the ID of the closest prototype and the frame number at which the cuboid was extracted. We trained our algorithm with values of *k* varying from 1 to 3. Here we report the performance obtained with the best setting for each class.

A performance comparison is shown in Table 4. The second column shows the result reported by Dollár et al. [46] using a 1-nearest neighbor classifier on histograms containing only words computed at spatial–temporal interest points. *1-NN* is the result obtained with the same method applied to histograms also including random points. *SVM* is the traditional SVM approach in which each video is represented by the histogram of words over the entire clip. The performance is measured using the F1 score which is defined as

$$F1 = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision}, \tag{21}$$

here we use this measure of performance instead of the ROC metric because the latter is designed for binary classification rather than detection tasks [55]. Our method achieves the best *F*1 score on all but one action.

4.2.3. Discriminative localization in human motion

For a qualitative evaluation of the ability to discover discriminative patterns in time series, we collected some accelerometer readings of human walking activity. A 40 Hz 3-axis accelerometer was attached to the left arm of a subject, and we collected a training set of 10 negative and 15 positive time series, respectively. The negative samples recorded the normal walking activity of the subject, while in each positive sample, the subject walked but fell twice during the course the activity. Each time series contains 2000 frames; at 40 Hz, this corresponds to 50 s. Some examples of the time series in this dataset are shown in Fig. 8.

We obtained a temporal codebook of 20 clusters using *K*-means on frame-level accelerometer vectors. Subsequently, each frame was represented by the ID of the cluster that it belonged to. We trained our algorithm and localized *k*-segmentations with values of *k* varying from 1 to 10. In Fig. 9, we show the qualitative results for discriminative localization in several time series that were not used in training. The proposed method correctly discovered the discriminative segments (falling events) for a wide range of *k* values.

4.3. Multi-class categorization of cooking activity

This section explores the use of accelerometers for activity classification in the context of cooking and preparing recipes in an unstructured environment. We performed our experiments on the Carnegie Mellon University Multimodal Activity (CMU-MMAC) database [56]. This collection contains multimodal measures of human subjects performing tasks involved in cooking five different recipes: brownies, scrambled eggs, pizza, salad, and sandwich. Fig. 10(a) shows an example of the data collection process, a subject is cooking scrambled eggs in a fully operable kitchen. Although the database contains multimodal measures (video, audio, motion capture, bodymedia, RFID, eWatch, IMUs), we only used the accelerometer readings from the five wired Inertial Measurement Units (IMUs). These 125 Hz accelerometers are triaxial and attached to the waist and the limbs of the subjects as shown in Fig. 10(b). We used the main dataset[7] which contains data of 39 subjects. We arbitrarily divided the data into disjoint training and testing subsets: subjects with odd IDs were used for training and subjects with even IDs were reserved for testing. The training and testing subsets contained 89 and 80 samples, respectively.

Previous work in the literature [57] has achieved high accuracy using acceleration data for classifying repetitive human activities such as walking, running, and bicycling. However, CMU-MMAC dataset is far more challenging because it was captured in an unstructured environment and the subjects were minimally instructed. As a consequence, how a recipe was cooked varied greatly from one subject to another. Moreover, the course of food preparation and recipe cooking contains a series of actions, and most of them are not repetitive. Many actions such as walking, opening the fridge, and turning on the oven are common for most recipes. More discriminative actions such as opening a brownie bag or cracking an egg are often buried in a long chain of actions.

We adopted the same feature representation as [57]. In particular, we computed a feature vector every second. To compute the feature vector at a specific time, we obtained a surrounding window of 1000 frames; at 125 Hz, this corresponds to 8 s. Mean, frequency-domain energy, frequency-domain entropy, and correlation features were extracted from this supporting window, as described in [57]. Every second of a time series was therefore associated with a feature vector of 150 dimensions. The attributes of these feature vectors were scale-normalized to have maximum magnitude of 1. These normalized

[7] http://kitchen.cs.cmu.edu/main.php

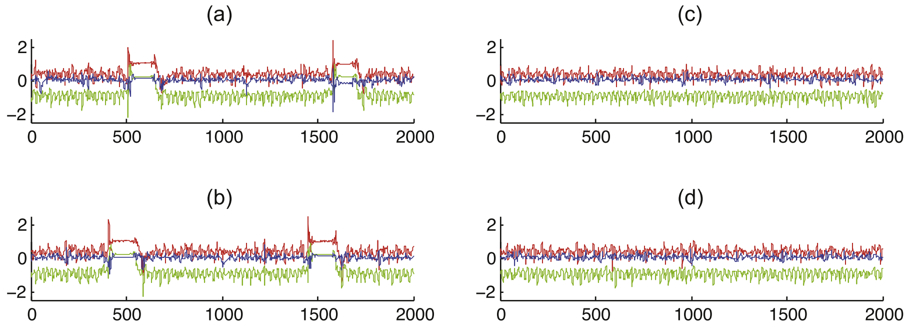

Fig. 8. Examples of accelerometer readings of human activity. Red, green, blue correspond to three channels of a triaxial accelerometer. Negative samples (c, d) recorded normal walking activity while positive samples (a, b) included the falling events. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

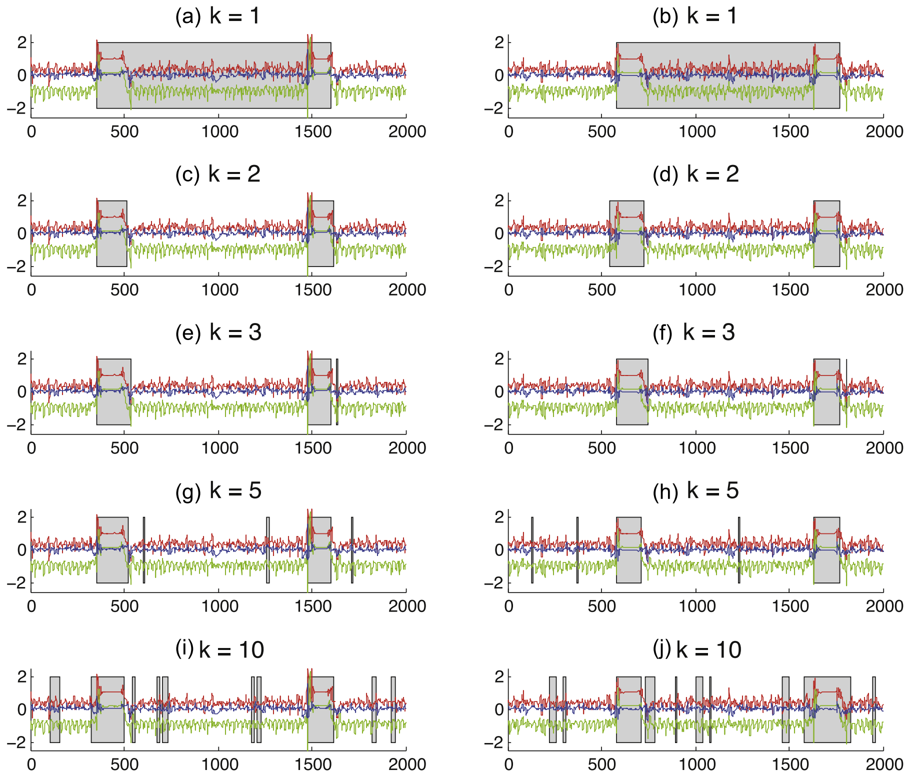

Fig. 9. Discriminative localization in human motion analysis. This figure shows two examples of testing time series and the results for different values of *k*, the number of segments in *k*-segmentations. The left sub-figures (a, c, e, g, i) show the same time series, while the right sub-figures (b, d, f, h, j) depict another time series. *k* is 1, 2, 3, 5, 10 for (a, b), (c, d), (e, f), (g, h), and (i, j), respectively. Our method successfully discovers the discriminative patterns (falling events) for a wide range of *k* values.

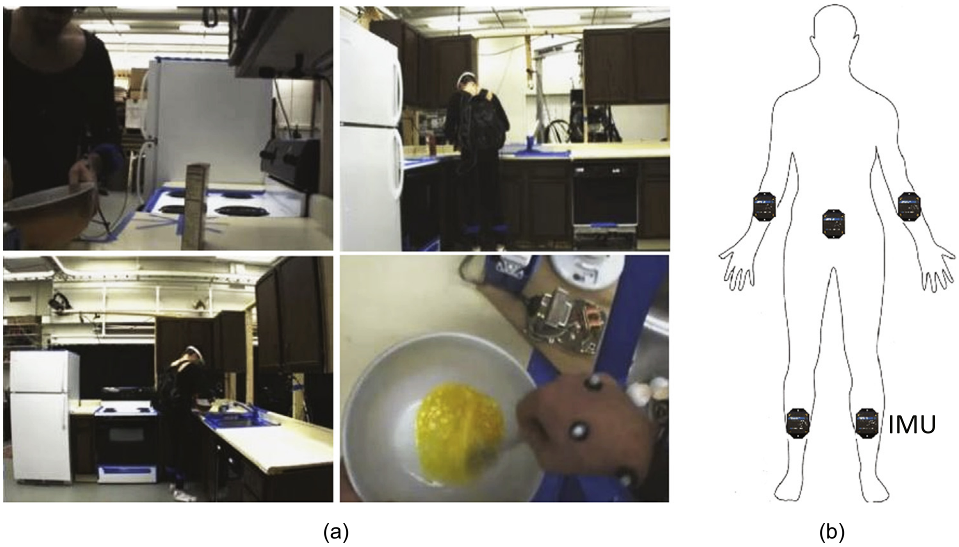

Fig. 10. CMU-MMAC dataset. (a) Data collection in action, a subject is cooking scrambled egg in a fully operable kitchen. (b) Locations of five wired Inertial Measurement Units (IMUs); the accelerometer readings of these IMUs are used for experiments in Section 4.3.

Table 5
Results on CMU-MMAC dataset: confusion matrix of the proposed method for five different recipes. The mean accuracy is 52.2%, compared with 42.4% from the traditional SVM. A random classifier would yield an expected accuracy of 20%.

	Brownie	Egg	Pizza	Salad	Sandwich
Brownie	68.8	6.2	6.2	0.0	18.8
Egg	25.0	31.2	12.5	12.5	18.8
Pizza	11.8	5.9	47.1	17.6	17.6
Salad	5.9	11.8	23.5	35.3	23.5
Sandwich	0.0	7.1	0.0	14.3	78.6

feature vectors were clustered using K-means to obtain a codebook of 50 temporal words. Subsequently, each second of the accelerometer data was represented by the ID of the closest temporal word. Because the amount of time to prepare and cook different recipes might differ, the histogram feature vector for a time series (either computed globally or on the foreground segments) was normalized by the length of the time series.

We implemented the multi-class categorization approach described in Section 3.3 combining with the multi-segment localization method of Section 3.2. In our implementation, k, the number of segments of k-segmentations, was set to 5. Table 5 displays the confusion matrix of this proposed method for categorizing five different recipes using accelerometer data. The mean accuracy is 52.2%. This is significantly higher than the mean accuracy of traditional SVM which is 42.4%. The expected accuracy of a random classifier is 20%.

5. Conclusions and future work

This paper proposed a novel framework for discriminative localization and classification from weakly labeled images or time series. We showed that the joint learning of the discriminative regions and of the region-based classifiers led to categorization accuracy superior to the performance obtained with supervised methods relying on costly human ground truth data. In future work we plan to investigate an unsupervised version of our approach for automatic discovery of object classes and actions from unlabeled collections of images and videos. Furthermore, we would like to extend our k-segmentation model to images in order to improve the recognition of objects having complex shapes.

Conflict of interest statement

None declared.

Acknowledgments

This work was partially supported by the National Science Foundation under Grant CPS-0931999. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Portions of this work were performed while Minh Hoai and Lorenzo Torresani were at Microsoft Research Cambridge.

The authors would like to thank Victor Lempitsky for useful discussion, Peter Gehler for pointing out related work, and Margara Tejera for helping with image annotation.

References

[1] S.X. Yu, J. Shi, Object-specific figure-ground segregation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2003.
[2] B. Leibe, B. Schiele, Interleaved object categorization and segmentation, in: Proceedings of the British Machine Vision Conference, 2003.
[3] E. Borenstein, E. Sharon, S. Ullman, Combining top-down and bottom-up segmentation, in: CVPR Workshop on Perceptual Organization in Computer Vision, 2004.
[4] Z. Tu, X. Chen, A. Yuille, S. Zhu, Image parsing: unifying segmentation, detection and recognition, International Journal of Computer Vision 63 (2) (2005) 113–140.
[5] N. Zlatoff, B. Tellez, A. Baskurt, Combining local belief from low-level primitives for perceptual grouping, Pattern Recognition 41 (4) (2008) 1215–1229.
[6] M. Hoai, Z.-Z. Lan, F. De la Torre, Joint segmentation and classification of human actions in video, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2011.
[7] P. Viola, M. Jones, Robust real-time face detection, International Journal of Computer Vision 57 (2) (2004) 137–154.
[8] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005.
[9] O. Chum, A. Zisserman, An exemplar model for learning object classes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2007.
[10] M.H. Nguyen, T. Simon, F. De la Torre, J. Cohn, Action unit detection with segment-based SVMs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010.
[11] M. Villamizar, J. Andrade-Cetto, A. Sanfeliu, F. Moreno-Noguer, Bootstrapping boosted random ferns for discriminative and efficient object classification, Pattern Recognition 45 (9) (2012) 3141–3153.
[12] C.H. Lampert, M.B. Blaschko, T. Hofmann, Beyond sliding windows: object localization by efficient subwindow search, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008.
[13] O. Maron, A. Ratan, Multiple-instance learning for natural scene classification, in: Proceedings of the International Conference on Machine Learning, 1998.
[14] S. Andrews, I. Tsochantaridis, T. Hofmann, Support vector machines for multiple-instance learning, in: Advances in Neural Information Processing Systems, 2003.
[15] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2003.
[16] B.C. Russell, A.A. Efros, J. Sivic, W.T. Freeman, A. Zisserman, Using multiple segmentations to discover objects and their extent in image collections, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2006.
[17] C. Rother, V. Kolmogorov, T. Minka, A. Blake, Cosegmentation of image pairs by histogram matching—incorporating a global constraint into MRFs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2006.
[18] D.S. Hochbaum, V. Sing, An efficient algorithm for co-segmentation, in: Proceedings of the International Conference on Computer Vision, 2009.
[19] S. Vicente, V. Kolmogorov, C. Rother, Cosegmentation revisited: Models and optimization, in: Proceedings of European Conference on Computer Vision, 2010.
[20] L. Cao, L. Fei-Fei, Spatial coherent latent topic model for concurrent object segmentation and classification, in: Proceedings of the International Conference on Computer Vision, 2007.
[21] S. Todorovic, N. Ahuja, Extracting subimages of an unknown category from a set of images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2006.
[22] C. Yang, T. Lozano-Pérez, Image database retrieval with multiple-instance learning techniques, in: International Conference on Data Engineering, 2000.
[23] Y. Chen, J.Z. Wang, Image categorization by learning and reasoning with regions, Journal of Machine Learning Research 5 (2004) 913–939.
[24] X. Qi, Y. Han, Incorporating multiple SVMs for automatic image annotation, Pattern Recognition 40 (2) (2007) 728–741.
[25] Y.-F. Li, J.T. Kwok, I.W. Tsang, Z.-H. Zhou, A convex method for locating regions of interest with multi-instance learning, in: Proceedings of the European Conference on Machine Learning, 2009.
[26] R.S. Cabral, F. De la Torre, J.P. Costeira, A. Bernardino, Matrix completion for multi-label image classification, in: Advances in Neural Information Processing Systems, 2011.
[27] C. Galleguillos, B. Babenko, A. Rabinovich, S. Belongie, Weakly supervised object recognition and localization with stable segmentations, in: Proceedings of the European Conference on Computer Vision, 2008.
[28] B. Babenko, M.-H. Yang, S. Belongie, Visual tracking with online multiple instance learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009.
[29] T. Deselaers, B. Alexe, V. Ferrari, Localizing objects while learning their appearance, in: Proceedings of the European Conference on Computer Vision, 2010.
[30] M. Blaschko, A. Vedaldi, A. Zisserman, Simultaneous object detection and ranking with weak supervision, in: Proceedings of Neural Information Processing Systems, 2010.
[31] M.H. Nguyen, L. Torresani, F. De la Torre, C. Rother, Weakly supervised discriminative localization and classification: a joint learning process, in: Proceedings of the International Conference on Computer Vision, 2009.

[32] H. Schweitzer, Utilizing scatter for pixel subspace selection, in: Proceedings of the International Conference on Computer Vision, 1999.

[33] M.H. Nguyen, F. De la Torre, Optimal feature selection for support vector machines, Pattern Recognition 43 (3) (2010) 584–591.

[34] P. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained, multiscaled, deformable part model, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[35] M.B. Blaschko, C.H. Lampert, Learning to localize objects with structured output regression, in: Proceedings of the European Conference on Computer Vision, 2008.

[36] P. Buehler, M. Everingham, A. Zisserman, Learning sign language by watching TV (using weakly aligned subtitles), in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[37] H. Zhong, J. Shi, M. Visontai, Detecting unusual activity in video, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2004.

[38] C. Fanti, L. Zelnik-Manor, P. Perona, Hybrid models for human motion recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[39] J.C. Niebles, H. Wang, L. Fei-Fei, Unsupervised learning of human action categories using spatial–temporal words, International Journal of Computer Vision 79 (3) (2008) 299–318.

[40] I. Tsochantaridis, T. Joachims, T. Hofmann, Y. Altun, Large margin methods for structured and interdependent output variables, Journal of Machine Learning Research 6 (2005) 1453–1484.

[41] C.-N.J. Yu, T. Joachims, Learning structural SVMs with latent variables, in: Proceedings of the International Conference on Machine Learning, 2009.

[42] P. Siva, C. Russell, T. Xiang, In defence of negative mining for annotating weakly labelled data, in: Proceedings of the European Conference on Computer Vision, 2012.

[43] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[44] J. Sivic, A. Zisserman, Video Google: a text retrieval approach to object matching in videos, in: Proceedings of the International Conference on Computer Vision, 2003.

[45] I. Laptev, T. Lindeberg, Space–time interest points, in: Proceedings of the International Conference on Computer Vision, 2003.

[46] P. Dollár, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: ICCV Workshop on Visual Surveillance & Performance Evaluation of Tracking and Surveillance, 2005.

[47] K. Crammer, Y. Singer, On the algorithmic implementation of multiclass kernel-based vector machines, Journal of Machine Learning Research 2 (2001) 265–292.

[48] Z. Fu, A. Robles-Kelly, J. Zhou, MILIS: multiple instance learning with instance selection, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (5) (2011) 958–977.

[49] X. Xu, B. Li, Evaluating multi-class multiple-instance learning for image categorization, in: Proceedings of the Asian Conference on Computer Vision, 2007.

[50] D. Nistér, H. Stewénius, Scalable recognition with a vocabulary tree, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2006.

[51] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, International Journal of Computer Vision 73 (2) (2001) 213–238.

[52] A. Vedaldi, B. Fulkerson, VLFeat: An Open and Portable Library of Computer Vision Algorithms, 2008, ⟨http://www.vlfeat.org/⟩.

[53] M. Grant, S. Boyd, Graph implementations for nonsmooth convex programs, in: V. Blondel, S. Boyd, H. Kimura (Eds.), Recent Advances in Learning and Control (A Tribute to M. Vidyasagar), Lecture Notes in Control and Information Sciences, Springer, 2008, pp. 95–110.

[54] M. Grant, S. Boyd, CVX: Matlab Software for Disciplined Convex Programming (Web Page & Software), October 2008, ⟨http://stanford.edu/~boyd/cvx⟩.

[55] S. Agarwal, A. Awan, D. Roth, Learning to detect objects in images via a sparse, part-based representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2004) 1475–1490.

[56] F. De la Torre, J. Hodgins, J. Montano, S. Valcarcel, J. Macey, Guide to the Carnegie Mellon University Multimodal Activity (CMU-MMAC) Database, Technical Report, CMU-RI-TR-08-22, Robotics Institute, Carnegie Mellon University, 2008.

[57] L. Bao, S.S. Intille, Activity recognition from user-annotated acceleration data, in: International Conference on Pervasive Computing, 2004.

Minh Hoai: Minh Hoai is a research fellow at Oxford University. He obtained Ph.D from Carnegie Mellon University in 2012 and B.E. from The University of New South Wales in 2005. He is the recipient of the CVPR2012 Best Student Paper Award.

Lorenzo Torresani:, Lorenzo Torresani is an Assistant Professor in theComputer Science Department at Dartmouth College.He received a Laurea Degree in Computer Science with summa cum laude honors from the University of Milan (Italy) in 1996, and an M.S. and a Ph.D. in Computer Science from Stanford University in 2001 and 2005, respectively. In the past, he has worked at several industrial research labs including Microsoft Research Cambridge, Like.com and Digital Persona. His research interests are in computer vision, machine learning and computer animation. In 2001, Torresani and his coauthors received the Best Student Paper Award at the IEEE Conference On Computer Vision and Pattern Recognition (CVPR). He is the recipient of a National Science Foundation CAREER Award.

Fernando De la Torre:, Fernando De la Torre is an Associate Research Professor in the Robotics Institute at Carnegie Mellon University. He received his B.Sc. degree in Telecommunications, as well as his M.Sc. and Ph. D degrees in Electronic Engineering from La Salle School of Engineering at Ramon Llull University, Barcelona, Spain in 1994, 1996, and 2002, respectively. His research interests are in the fields of Computer Vision and Machine Learning. Currently, he is directing the Component Analysis Laboratory (http://ca.cs.cmu.edu) and the Human Sensing Laboratory (http://humansensing.cs.cmu.edu) at Carnegie Mellon University. He has over 150 publications in referred journals and conferences and is Associate Editor at IEEE PAMI. He has organized and co-organized several workshops and has given tutorials at international conferences on the use and extensions of Component Analysis.

Carsten Rother: Carsten Rother received his Diploma degree with distinction in 1999 at the University of Karlsruhe, Germany. He did his PhD at the Royal Institute of Technology Stockholm, Sweden. From 2003 to 2013, he was PostDoc, Researcher and then Senior Researcher with Microsoft Research Cambridge. Since 2013 he is full Professor at TU Dresden running the Computer Vision Lab Dresden (CVLD). His research interests are in the field of Markov Random Field Models, low-level vision, such as segmentation and stereo, and Vision for Graphics. He has co-authored more than 100 articles and has an H-index of 40. He won six best paper (honourable) mention awards and received in 2009 the Olympus Award of the German Society of pattern recognition (DAGM), which is the highest award for young scientists in the field of computer vision. He co-edited a book on Markov Random Fields for Vision and Image Processing, MIT Press 2011. He is associated editor for TPAMI, and has been area chair and reviewer for many major conferences in the field.