ABSTRACT

Title of dissertation: GENERATING DISCRIMINATIVE OBJECT PROPOSALS VIA SUBMODULAR RANKING

Yangmuzi Zhang, Doctor of Philosophy, 2016

Dissertation directed by: Professor Larry S. Davis
Department of Electrical and Computer Engineering

Object recognition has long been a core problem in computer vision. To improve object spatial support and speed up object localization for object recognition, generating high-quality category-independent object proposals as the input for object recognition system has drawn attention recently. Given an image, we generate a limited number of high-quality and category-independent object proposals in advance and used as inputs for many computer vision tasks. Image classification is one of the most fundamental task in computer vision. We design an efficient dictionary-based model for image classification. We further extend the work to a discriminative dictionary learning method for tensor sparse coding. Activity classification is another challenging task in computer vision system. To address this problem, we propose a semi-parametric method to model crowded scenes for abnormal activity detection.

In the first part, a multi-scale greedy-based object proposal generation approach is presented. Based on the multi-scale nature of objects in images, our approach is built on top of a hierarchical segmentation. We first identify the representative and diverse exem-

plar clusters within each scale by using a diversity ranking algorithm. Object proposals are obtained by selecting a subset from the multi-scale segment pool via maximizing a submodular objective function, which consists of a weighted coverage term, a single-scale diversity term and a multi-scale reward term. The weighted coverage term forces the selected set of object proposals to be representative and compact; the single-scale diversity term encourages choosing segments from different exemplar clusters so that they will cover as many object patterns as possible; the multi-scale reward term encourages the selected proposals to be discriminative and selected from multiple layers generated by the hierarchical image segmentation. The experimental results on the Berkeley Segmentation Dataset and PASCAL VOC2012 segmentation dataset demonstrate the accuracy and efficiency of our object proposal model. Additionally, we validate our object proposals in simultaneous segmentation and detection and outperform the state-of-art performance.

To classify objects in the image, we design a discriminative, structural low-rank framework for image classification. We use a supervised learning method to construct a discriminative and reconstructive dictionary. By introducing an ideal regularization term, we perform low-rank matrix recovery for contaminated training data from all categories simultaneously without losing structural information. A discriminative low-rank representation for images with respect to the constructed dictionary is obtained. With semantic structure information and strong identification capability, this representation is good for classification tasks even using a simple linear multi-classifier.

In the third part, a novel approach to learn a discriminative dictionary over a tensor sparse model is presented. A structural incoherence constraint between dictionary atoms from different classes is introduced to promote discriminating information into the

dictionary. The incoherence term encourages dictionary atoms to be as independent as possible. In addition, we incorporate classification error into the objective function of dictionary learning. The dictionary is learned in a supervised setting to make it useful for classification. A linear multi-class classifier and the dictionary are learned simultaneously during the training phase. Our approach is evaluated on three types of public databases, including texture, digit, and face databases. Experimental results demonstrate the effectiveness of our approach.

In the final part, we present a fully unsupervised method for abnormal activity detection in crowded scenes. Neither normal nor abnormal training samples are needed before the detection. In crowded scenes, normal activities are the behaviours performed by majority of people and abnormalities are behaviours that occur rarely and are different from most others. We present a scan statistic method to capture abnormality. A semi-parametric density ratio method is used to model the observations. We successfully apply our algorithm to detect abnormal activities in different scenarios. Our approach achieves performance that is competitive to other state-of-the-art supervised approaches.

GENERATING DISCRIMINATIVE OBJECT PROPOSALS VIA SUBMODULAR RANKING

by

Yangmuzi Zhang

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2016

Advisory Committee:
Professor Larry S. Davis, Chair/Advisor
Professor Behtash Babadi
Professor Ramani Duraiswami
Professor Piya Pal
Professor Min Wu

Acknowledgments

I owe my gratitude to all the people who have made this thesis possible and because of whom my graduate experience has been one that I will cherish forever.

First and foremost I'd like to thank my advisor, Professor Larry S. Davis for giving me an invaluable opportunity to work on challenging and extremely interesting projects over the past five years. He has always made himself available for help and advice and there has never been an occasion when I've knocked on his door and he hasn't given me time. It has been a pleasure to work with and learn from such an extraordinary individual.

Thanks are due to Professor Behtash Babadi, Professor Ramani Duraiswami, Professor Piya Pal and Professor Min Wu for agreeing to serve on my thesis committee and for sparing their invaluable time reviewing the manuscript.

My colleagues at the computer vision laboratory have enriched my graduate life in many ways and deserve a special mention. My interaction with Dr. Zhuolin Jiang, Dr. Yang Hu, Dr. Xi Chen has been very fruitful.

I owe my deepest thanks to my family - my mother and father who have always stood by me and guided me through my career, and have pulled me through against impossible odds at times. Words cannot express the gratitude I owe them. I'd like to express my gratitude to my friends at my place of residence, Jiaying He, Min Zhou, and Daimeng Zhang for their friendship and support.

It is impossible to remember all, and I apologize to those I've inadvertently left out. Thank you all!

Table of Contents

List of Figures

List of Abbreviations

α alpha
β beta

Chapter 1: Introduction

1.1 Overview

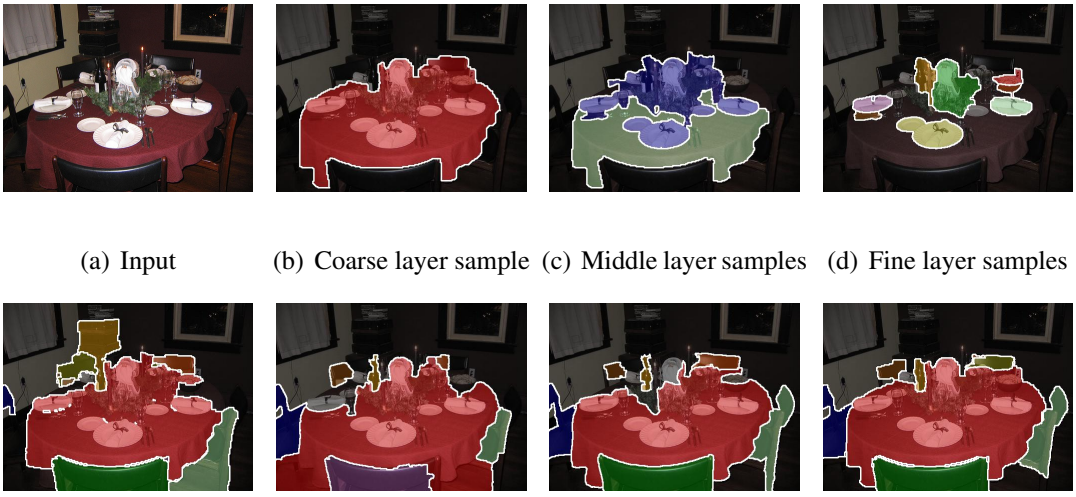
Object recognition has long been a core problem in computer vision. Recent developments in object recognition provide two effective solutions: 1) sliding-window-based object detection and localization [4–6], 2) segmentation-based approaches [1–3, 7]. The sliding window approach incurs high computational cost as it analyses windows over a very large set of locations and scales. Segmentation-based methods lead to fewer regions to consider and to better spatial support for objects of interest with richer shape and contextual information; but the problem of segmenting an image to identify regions with high object spatial support is a challenge.

To improve object spatial support and speed up object localization for object recognition, generating high-quality category-independent object proposals as the input for object recognition system has drawn attention recently [2, 3, 7, 8]. Motivated by findings from cognitive psychology and neurobiology [9–12] that the human vision system has the amazing ability to localize objects before recognizing them, a limited number of high-quality and category-independent object proposals can be generated in advance and used as inputs for many computer vision tasks. This approach has played a dominant role in semantic segmentation [13, 14] and leads to competitive performance on detection [15].

There are two main categories of object proposal generation methods depending on the shape of proposals: bounding-box-based proposals [7, 8, 16] and segment-based proposals [2, 3, 17].

Objects in an image are intrinsically hierarchical and of different scales. Consider the table in Figure 1.1(a) for example. The objects on the table can be regarded as a part of the table (Figure 1.1(b)), and at the same time, they constitute a group of objects on the table (Figure 1.1(c)). More specifically, these objects include plates, forks, the Santa Claus, and a bottle (Figure 1.1(d)). Therefore, multi-scale segmentation is essential to localize and segment different objects. There have been a few attempts [1–3] to combine multiple scale information in the object proposal generation process, but very few papers have studied the importance of proposal selection given segments from hierarchical image segmentations. Figure 1.1(e)1.1(f)1.1(g) show the generated proposals from three state-of-art algorithms [1–3]. However, they do not cover all the objects in the image well.

We present a greedy approach to efficiently extract high-quality object proposals from an image via maximizing a submodular objective function. We first construct diverse exemplar clusters of segments over a range of scales using diversity ranking; then rank and select high-quality object proposals from the multi-scale segment pool generated by hierarchical image segmentation. Our objective function is composed of three terms: a weighted coverage term, a single-scale diversity term and a multi-scale reward term. The first term encourages the selected set to be compact and well represent all segments in an image. The second term enforces the selected segments (object proposals) to be diverse and cover as many different objects as possible. The third term encourages the selected proposals to correspond to objects with high confidence and selected from differ-



(a) Input (b) Coarse layer sample (c) Middle layer samples (d) Fine layer samples

(e) CPMC [1] (f) Categ. Indep. [2] (g) MCG [3] (h) Our method

Figure 1.1: Objects in an image are naturally hierarchical. (a) is an original image from Pascal VOC2012; (b) - (d) show segments around the table from different scales using method [1]; (e) shows the best seven object proposals generated from CPMC [1]; (f) are proposals from Categ. Indep. [2]; (g) are proposals from MCG [3]; (h) are proposals from our method.

ent scales. The algorithm takes object scale information into account and avoids selecting segments from the same layer repeatedly. Compared to existing segment-based methods, our method (Figure 1.1(h)) can select representative, diverse and discriminative object proposals from different layers (for example, the bottle from fine layer and the table from coarse layer).

1.2 Related Work

The goal of object proposal algorithms is to generate a small number of high-quality category-independent proposals such that each object in an image is well captured by at least one proposal [2, 18]. Existing object proposal approaches can be roughly divided into bounding-box and segment based approaches. [16] generated bounding boxes by utilizing edge and contour clues. In [7], a data-driven grouping strategy which combines segmentation and exhaustive search is presented to produce bounding-box-based proposals. [8] proposed the binarized normed gradients (BING) feature to efficiently produce object boxes. Instead of generating bounding-box-based proposals, our work focuses on extracting segment-based proposals which aims to cover all the objects in an image and can provide more accurate shape and location information. Some algorithms have been reported to generate segment-based object proposals. [1] segmented objects by solving a series of constrained parametric min-cut (CPMC) problems. [19] reused inference in graph cuts to solve the parametric min-cut problems much more efficiently. [2] performed graph cuts and ranked proposals using structured learning. In [3], a hierarchical segmenter is used to combine multi-scale information, and a grouping strategy is presented to extract

object candidates. Different from their work, we design an efficient greedy-based ranking method to leverage multi-scale information in the process of selecting object proposals from a large hierarchical segment pool.

Object proposals have been used in many computer vision tasks, such as segmentation [1,13], object detection [15] and large-scale classification [7]. Semantic segmentation and object detection have been shown to support each other mutually in a wide variety of algorithms. [20] showed that better quality segmentation can improve object recognition performance. [15,21,22] used hierarchical segmentations and combined several top-down cues for object detection. The more demanding task of simultaneous detection and segmentation (SDS) is investigated in [22] which detects and labels the segments at the same time. We use this same detection and segmentation framework but with our object proposal generation method to demonstrate the effectiveness of proposals generated by our approach.

Submodular optimization is a useful optimization tool in machine learning and computer vision problems [23–28]. [23] demonstrates how submodularity speeds up optimization algorithm in large scale problems. In [25], a diffusion-based framework is proposed to solve cosegmentation problems via submodular optimization. [26] used the facility location problem to model salient region detection where salient regions are obtained by maximizing a submodular objective function.

1.3 Preliminaries

Submodularity: Let V be a finite set, $A \subseteq B \subseteq V$ and $a \in V \setminus B$. A set function $F : 2^v \rightarrow R$ is submodular if $F(A \bigcup a) - F(A) \geqslant F(B \bigcup a) - F(B)$. This is the diminishing return property: adding an element to a smaller set helps more than adding it to a larger set [29].

Theorem 1. Given functions $F : 2^V \rightarrow R$ and $f : R \rightarrow R$, the composition $F' = f \circ F : 2^V \rightarrow R$ is non-decreasing submodular, if f is non-decreasing concave and F is non-decreasing submodular.

1.4 Submodular Proposal Extraction

We first obtain a large pool of segments from different scales using hierarchical image segmentation. Diverse exemplar clusters are then generated via diversity ranking within each layer to discover potential objects in an image. We define a submodular objective function to rank and select a discriminative and compact subset from a large set of segments of different scales, then the selected segments are used as the final object proposals.

1.4.1 Hierarchical Segmentation

We build our object proposal generation framework on top of hierarchical segmentation. Following [1,19], we generate segments for an image at different scales by solving multiple constrained parametric min-cut problems with different seeds and unary terms.

1.4.2 Exemplar Cluster Generation

In a coarser layer, an image is segmented into only a few segments. However, the number of segments increases dramatically as we go to finer layers. To reduce the redundancy and maintain segment diversity, we introduce an exemplar cluster generation step to pre-process segments within layers.

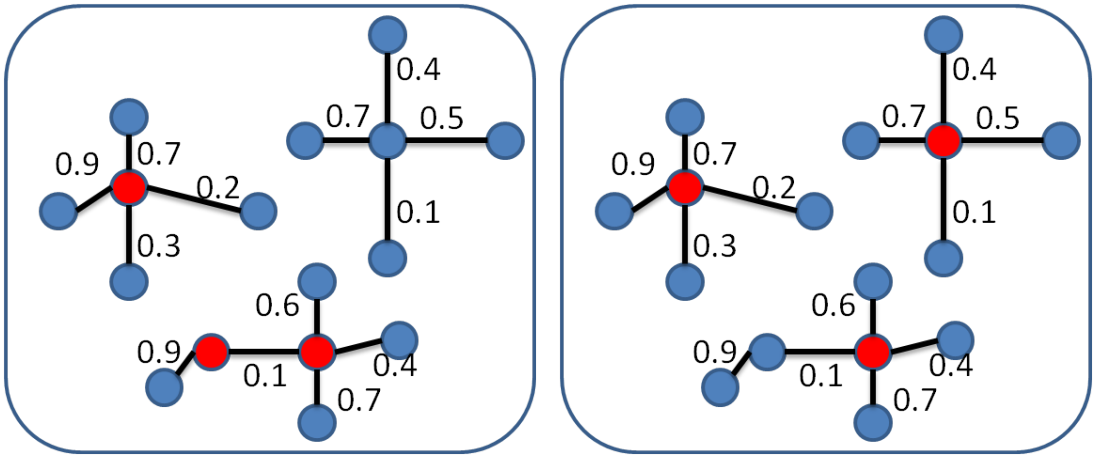
Let V denote the set containing segments from all layers of an image (the multi-scale segment pool), and V^l be the set of segments from layer l. Then $V = \bigcup_{l=1}^{L} V^l$, L is the total number of layers, and V^ls are disjoint. For each layer l, we obtain a partition of its segments $\{P_1^l, P_2^l, ..., P_t^l\}$ using a diversity ranking algorithm [25]. P_t^l is the set of segments assigned to cluster t. Each segment belongs to only one cluster, and clusters are disjoint. For each layer L, we have $V^l = \cup_{t=1}^{T} P_t^l$, where T is the number of clusters[1].

1.4.3 Submodular Multi-scale Proposal Generation

We present a proposal generation method by selecting a subset A which contains high-quality segments (object proposals) from the set V.

Given an image I, we construct an undirected graph $G = (V, E)$ for the segment hypotheses in I. Each vertex $v \in V$ is an element from the multi-scale segment pool. Each edge $e \in E$ models the pairwise relation between vertices. Two segments are connected if they are overlapping (between layers) or adjoining (within a layer). The weight w_{ij} associated with the edge e_{ij} measures the appearance similarity between vertices v_i and v_j. We extract a CNN feature descriptor [30] using VGGNet as the pre-trained model

[1]For coarser layer, T is the number of initial segments obtained from hierarchical segmentation.



(a) H(A)=7.7 (b) H(A)=8.6

Figure 1.2: The weighted coverage term for the representative proposal selection (best viewed in color). The node denotes the segment vertex, and the value next to the edge is the similarity between vertices. The red nodes are selected vertices. To select three nodes among all, by computing the weighted coverage term, we favours selecting a more representative set (three center nodes in (b) will lead to higher $H(A)$ than the less representative one since the two nodes are from one group in (a)). Hence the selected A is representative and compact.

for each segment: $X = [x_1, x_2, ..., x_{|V|}]$. w_{ij} is defined as the Gaussian similarity between two vertices' feature descriptors. As suggested in [31], we set the normalization factor $\epsilon = 1/\sigma_i\sigma_j$ and the local scale σ_i is selected by the local statistic of vertex i's neighbourhood. We adopt the simple choice which sets $\sigma_i = d(x_i, x_M)$ where x_M corresponds to the M'th closest neighbour of vertex i.

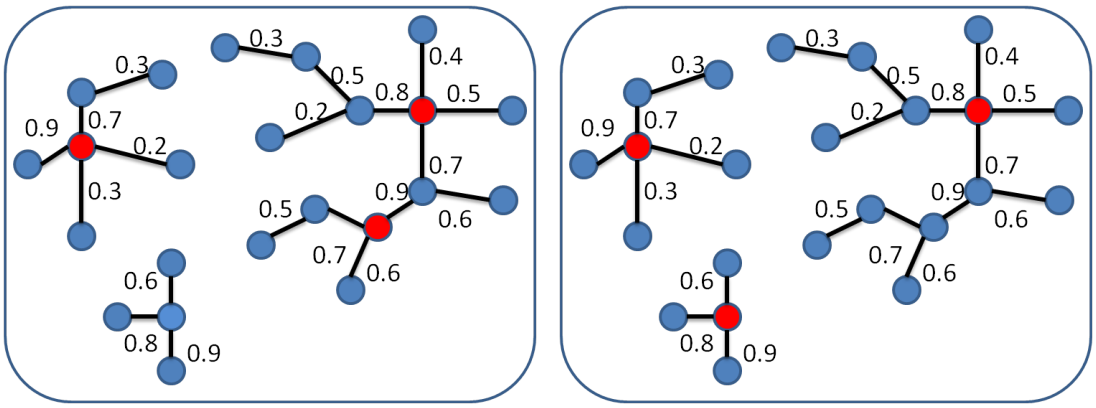
1.4.3.1 Weighted Coverage Term

The selected subset A should be representative of the whole set V. The similarity of subset A to the whole set V is maximized with a constraint on the size of A. Accordingly, we introduce a weighted coverage term for selecting representative proposals.

Let N_A denote the number of selected segments. Then the weighed coverage term is formulated as:

$$H(A) = \sum_{i \in V} \max_{j \in A} w_{ij} \tag{1.1}$$
$$s.t. \quad A \subseteq V, N_A \leqslant K$$

where K is the maximum number of segments to be chosen in set A. The weighted coverage of each segment v_i is $\max_{j \in A} w_{ij}$. Equation (2.5) measures the representativeness of A to V and favours selecting segments which can cover (or represent) the other unselected segments. Maximizing the weighted coverage term encourages the selected set A to be representative and compact as shown in Figure 1.2.



(a) D(A)=0.90 (b) D(A)=1.13

Figure 1.3: The single-layer diversity term for the diverse proposal selection. Each node denotes a segment vertex (best viewed in color). Similarity between vertices are labelled next to each edge. The red node labels the selected segments. Each figure shows three exemplar clusters as connected groups. We can see the three exemplar clusters are unbalanced. Purely computing the weighted coverage term will pick the third node from the largest cluster to gain more similarity between the selected set and the whole set as in (a). While by computing the single-layer diversity term, we observe that (b) is preferred to (a) as it encourage diversity among the selected nodes.

1.4.3.2 Single-Scale Diversity Term

The weighted coverage term will give rise to a highly representative set A; however, segments from each layer (corresponding to each image scale) still possess redundancy. Therefore, we introduce a diversity term to force segments within a layer l to be different. The single-layer diversity term is formulated as follows:

$$D(A) = \sum_{l=1}^{L} D_l(A) = \sum_{t,l} \sqrt{\sum_{j \in P_t^l \cap A} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} \tag{1.2}$$

where P_t^l is the set of segments which belong to cluster t in layer l (defined in section 1.4.2). $|V^l|$ is the number of segments in layer l. This single-scale diversity term encourages A to include elements from different clusters and leads to more diverse segments from each layer. The single-layer diversity term is submodular; a detailed proof is provided in the Appendix A.

In many images, the background composes a large part of the image. For a single layer, the segments corresponding to objects are only a small percentage of all segments. The segment distributions corresponding to different objects and the background are generally unbalanced. The weighted coverage term favours selecting segments that well represent all segments, resulting in redundancy and occasionally missing small objects. Together with the single-layer diversity term, diversity among the selected segments are enforced as shown in Figure 1.3.

1.4.3.3 Multi-Scale Reward Term

Considering the multi-scale nature of objects in an image, we propose the following discriminative multi-scale reward term to encourage selected segments to have high likelihood of high object coverage. The multi-scale reward term is defined as:

$$R(A) = \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap A} r_j} \tag{1.3}$$

V^l is the set of segments from layer l. The value r_j estimates the likelihood of a segment to be an object. It determines the priority of a segment being chosen in its layer. We use

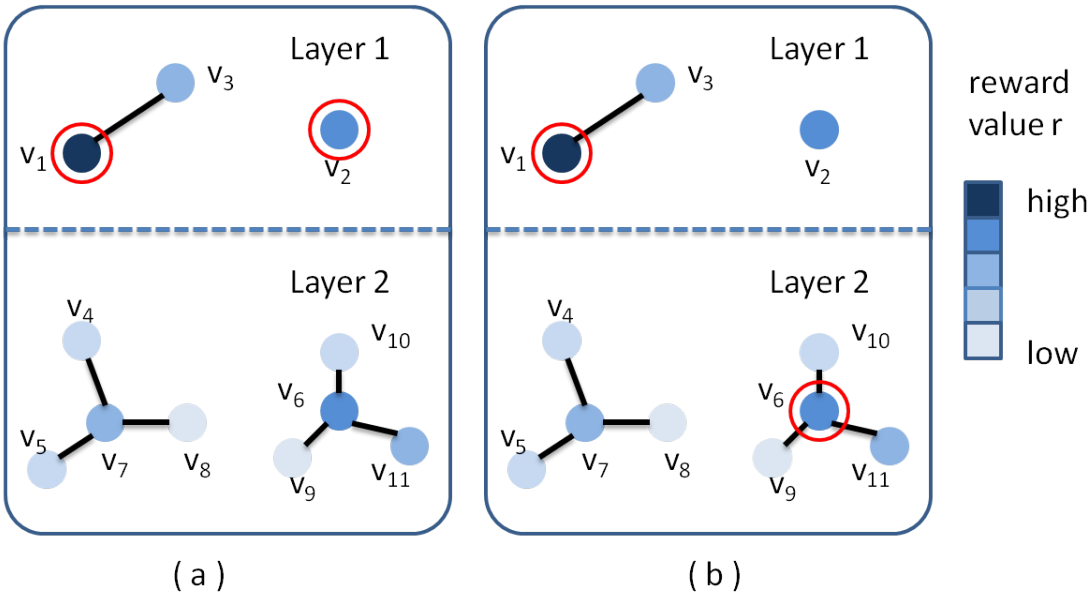


Figure 1.4: The multi-scale reward term for selecting proposals from different scales (best viewed in color). The nodes represent segments. The reward value r_i of segment v_i is reflected by color. The higher r_i, the more likely it is an object. The red circle denotes the selected nodes. Suppose v_1 has already been selected. We observe that $R\{v_1, v_2\} - R\{v_1\} < R\{v_1, v_6\} - R\{v_1\}$. In another word, although v_2 and v_6 have similar reward value, v_6 from layer 2 will brings higher marginal gain; thus v_6 is favoured over v_2 and (b) is preferred to (a).

CNN features to train a SVM model over object segments and non-object segments in training images and then assign a confidence score for each segment during testing. The confidence score is used as r_j for a segment v_j.

The multi-scale reward term encourages A to select a set of discriminative segments from multi-scale segments generated from a hierarchical segmentation. As soon as an element is selected from a layer, other elements from the same layer start to have di-

minishing gain because of the submodular property of $R(A)$. A simple example is shown in Figure 1.4. Similar to $D(A)$, $R(A)$ is submodular and the proof is presented in the Appendix A.

1.5 Optimization

We combine the weighted coverage term, the single-scale diversity term and the multi-scale reward term to find high-quality object proposals. The final objective function of object proposal generation is formulated as below:

$$\begin{aligned} \max_{A} F(A) &= \max_{A} H(A) + \alpha D(A) + \beta R(A) \\ &= \max_{A} \sum_{i \in V} \max_{j \in A} w_{ij} + \beta \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap A} r_j} \\ &\quad + \alpha \sum_{n,l} \sqrt{\sum_{j \in P_t^l \cap A} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} \\ &\quad s.t. \quad A \subseteq V, N_A \leq K, \alpha \geq 0, \beta \geq 0 \end{aligned} \tag{1.4}$$

The submodularity is preserved by taking non-negative linear combinations of the three submodular terms $H(A)$, $D(A)$, and $R(A)$. Direct maximization of equation (A.4) is an NP-hard problem. We can approximately solve the problem via a greedy algorithm [29, 32] based on its submodularity property. A lower bound of $(e-1)/e$ times the optimal value is guaranteed as proved in [29] (e is the base of the natural logarithm).

The algorithm starts from an empty set $A = \varnothing$. It adds the element a^* which provides the largest marginal gain among the unselected elements to A iteratively. The iterations stop when $|A|$ reaches the desired capacity number K. The optimization steps can be further accelerated using a lazy greedy approach from [23]. Instead of recomputing

gain for every unselected element after each iteration, an ordered list of marginal benefits will be maintained in descending order. Only the top unselected segment is re-evaluated at each iteration. Other unselected segments will be re-evaluated only if the top segment does not remain at the top after re-evaluation. The pseudo code is presented in Algorithm 1.

Algorithm 1 Submodular object proposal generation

Input: I, $G = (V, E)$, K, α, β

Output: A

Initialization: $A \leftarrow \varnothing$, $U \leftarrow V$

loop

$a = arg a \in U \max F(A \cup \{a\}) - F(A)$

if $|A| \geq K$ **then**

break

$A \leftarrow A \cup \{a^*\}$

$U \leftarrow U - \{a^*\}$

	AUC	Recall	BSS
C,T+layout [2]	77.5	83.4	67.2
all feature [2]	80.2	79.7	66.2
Ours	81.1	83.6	71.8

Table 1.1: Comparison of object proposals' quality on the BSDS dataset, measured with AUC, recall and BSS.

Method	N	Plane	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	MBike	Person	Plant	Sheep	Sofa	Train	TV	Global
Ours	1100	**82.3**	48.8	**84.6**	**76.7**	**71.4**	**80.6**	67.7	**93.1**	**69.7**	**86.0**	78.5	**89.7**	**83.2**	77.3	72.9	**70.4**	77.8	**85.8**	**85.0**	**87.5**	**76.5**
[3]	1100	80.0	47.8	83.9	76.4	71.1	78.5	**68.9**	89.3	68.5	85.9	79.8	85.8	80.4	75.4	**73.5**	69.3	**84.9**	82.6	81.7	85.8	76.0
[2]	1100	75.1	**49.1**	80.7	68.8	62.8	76.4	63.3	89.4	64.6	83.0	**80.3**	83.7	78.4	**78.0**	66.9	66.2	69.5	82.0	84.3	81.8	71.6
[13]	1100	74.4	46.6	80.5	69.4	64.6	73.5	61.2	89.0	65.1	80.5	78.4	85.2	77.2	70.6	67.9	68.8	73.5	81.6	75.8	82.0	71.4
[35]	1100	73.8	40.6	75.8	66.7	52.7	79.7	50.6	91.2	59.2	80.2	80.7	87.4	79.0	74.7	62.1	54.6	65.0	84.6	82.4	79.5	67.4
[36]	1100	68.3	39.6	70.6	64.8	58.0	68.2	51.8	77.6	58.2	72.6	70.4	74.0	66.2	59.9	59.8	55.4	67.7	71.3	68.6	78.7	63.1
ours	100	**75.2**	**40.8**	**78.4**	**70.3**	**55.5**	**72.8**	**51.1**	**83.4**	**56.8**	77.3	66.7	**84.4**	**75.2**	65.9	59.3	**54.9**	68.1	**77.9**	**76.1**	76.8	**64.3**
[3]	100	70.2	38.8	73.6	67.7	55.3	68.5	50.6	82.4	54.4	**78.1**	67.7	77.7	69.3	66.3	**59.9**	51.4	**70.2**	74.1	72.6	**78.1**	63.7
[2]	100	70.6	**40.8**	74.8	59.9	49.6	65.4	50.4	81.5	54.5	74.9	**68.1**	77.3	69.3	**66.8**	56.2	54.3	64.1	72.0	71.6	69.9	61.7
[1]	100	72.7	36.2	73.6	63.3	45.4	67.4	39.5	84.1	47.7	73.2	64.0	81.1	72.2	64.3	52.8	42.9	62.2	72.2	74.3	69.5	59.0

Table 1.2: VOC2012 val set. Jaccard index at the instance level and class level.

1.6 Experiments

We evaluate our approach on two public datasets: BSDS [33] and PASCAL VOC2012 [34] segmentation dataset. The results for PASCAL VOC2012 are on the validation set of the segmentation task. We evaluate the object proposal quality by assessing the best proposal for each object using the Jaccard index score (see details in section 1.6.1). We also compare our ranking method with several baselines [2] and analyses the efficiency of our object proposals on the object recognition task.

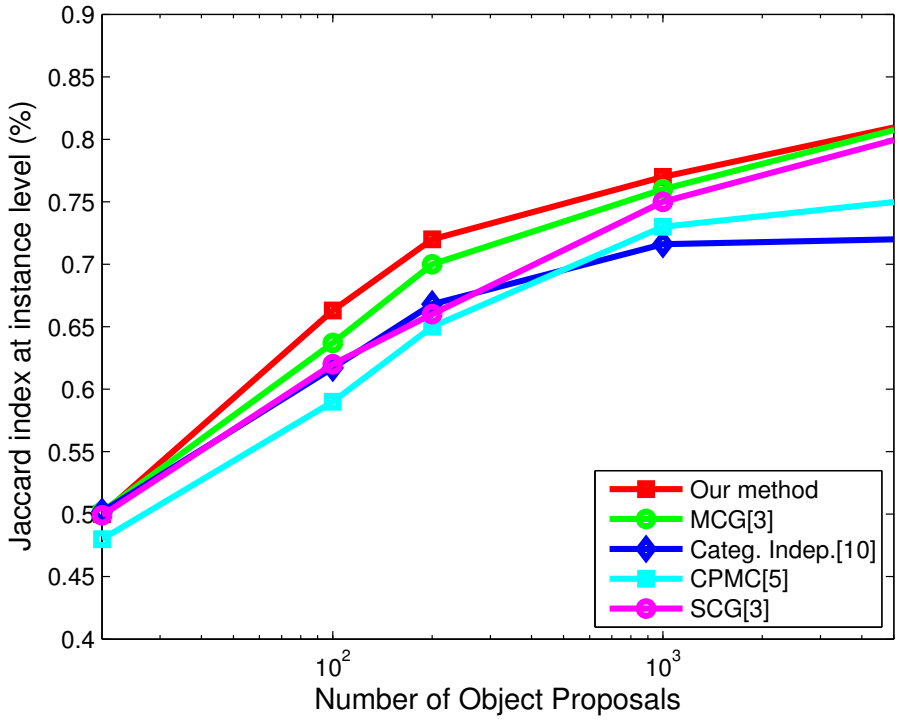


Figure 1.5: Object proposal quality on PASCAL VOC2012 validation set, measured with the Jaccard index at instance level J_i.

Figure 1.6: Sample object proposals from the PASCAL VOC2012. The left column shows the best four proposals for objects in our model. The remaining columns show the highest ranked proposals with at least 50 percent overlap with an object. The second column is from our method, the third column is from Categ. Indep. [2], the fourth column is from CPMC [1], and the last column is from MCG [3].

1.6.1 Proposal evaluation

To measure the quality of a set of object proposals, we followed [3] and compute the Jaccard index score, or the best segmentation overlap score (BSS) for each object. The overall quality of a object proposal set is measured at the class level and the instance level. The Jaccard index at instance level, denoted as J_i, is defined as the mean of BSS over all objects. The Jaccard index at class level, J_c is defined as the mean of BSS over objects from each category.

1.6.1.1 BSDS dataset

We compare our object proposals with [2]. For fair comparison, we also compute the area under the ROC curve (AUC) and recall defined with an overlap threshold at 50 per cent. The results are summarized in Table 1.1. Our object proposal achieves the best performance.

1.6.1.2 PASCAL VOC2012

We evaluate our object proposal approach on the PASCAL VOC2012 validation dataset. The SVM classifier for reward value (details in section 2.2.2.3) is trained on the training dataset. Our object proposals are compared with [1–3, 13, 35, 36]. As shown in Table 1.2, our method outperform all other methods with the same number of object proposals for Jaccard index at the instance level. Meanwhile, we achieve the highest scores on most of the classes (14 out of 20). In Figure 1.5, we show how J_i changes as the number of object proposals increases. Since our approach prefers to select representa-

Method	Plane	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	MBike	Person	Plant	Sheep	Sofa	Train	TV	Mean
O_2P [22]	56.5	19.0	23.0	12.2	11.0	48.8	26.0	43.3	4.7	15.6	7.8	24.2	27.5	32.3	23.5	4.6	32.3	20.7	38.8	32.3	25.2
SDS-A [22]	61.8	43.4	46.6	27.2	28.9	61.7	46.9	58.4	17.8	38.8	18.6	52.6	44.3	50.2	48.2	23.8	54.2	26.0	53.2	55.3	42.9
SDS-B [22]	65.7	**49.6**	47.2	30.0	31.7	**66.9**	50.9	69.2	19.6	42.7	22.8	56.2	51.9	52.6	52.6	25.7	**54.2**	**32.2**	59.2	58.7	47.0
SDS-C [22]	67.4	**49.6**	49.1	29.9	32.0	65.9	**51.4**	**70.6**	**20.2**	42.7	22.9	58.7	54.4	53.5	54.4	24.9	54.1	31.4	62.2	59.3	47.7
Ours	**68.2**	14.0	**64.7**	**51.3**	**39.3**	62.1	45.6	65.8	9.9	**49.1**	**30.8**	**61.9**	**54.9**	**65.9**	**54.5**	**31.8**	48.4	29.5	**73.9**	**65.6**	**48.9**

Table 1.3: Results on AP^r on the PASCAL VOC2012 val. All numbers are %.

Method	Plane	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	MBike	Person	Plant	Sheep	Sofa	Train	TV	Mean
O_2P [22]	46.8	21.2	22.1	13.0	10.1	41.9	24.0	39.2	6.7	14.6	9.9	24.0	24.4	28.6	25.6	7.0	29.0	18.8	34.6	25.9	23.4
SDS-A [22]	48.3	39.8	39.2	25.1	26.0	49.5	39.5	50.7	17.6	32.5	18.5	46.8	37.7	41.1	43.2	23.4	43.0	26.2	45.1	47.7	37.0
SDS-B [22]	51.1	42.1	40.8	27.5	26.8	53.4	42.6	56.3	18.5	36.0	20.6	48.9	41.9	43.2	45.8	24.8	44.2	29.7	48.9	48.8	39.6
SDS-C [22]	53.2	42.1	42.1	27.1	27.6	53.3	42.7	57.3	19.3	36.3	21.4	49.0	43.6	43.5	47.0	24.4	44.0	29.9	49.9	49.4	40.2
SDS-C+ref [22]	52.3	**42.6**	42.2	28.6	28.6	**58.0**	**45.4**	58.9	**19.7**	37.1	22.8	49.5	42.9	45.9	**48.5**	25.5	**44.5**	**30.2**	52.6	51.4	41.4
Ours	**54.7**	19.4	**54.3**	**40.9**	**34.4**	52.0	41.3	**59.3**	13.3	**42.9**	**25.8**	**51.9**	**44.8**	**51.5**	47.0	**31.4**	42.6	28.5	**59.2**	**53.8**	**42.4**

Table 1.4: Results on AP^r_{vol} on the PASCAL VOC2012 val. All numbers are %.

tive, diverse and multi-scale object proposals, our proposal quality outperform MCG [3], Categ. Indep. [2], CPMC [1], and SCG [3] with only a small number of proposals. In Figure 1.6, we show some qualitative results of our object proposals. We observe that our proposals can capture diverse objects of different sizes. In addition, we compare our proposal generation time with MCG [3] which also uses multi-scale information. Our method takes about 7 seconds per image compared to 10 seconds reported in [3]. The parameters are set $\alpha = 3.9$, $\beta = 2.0$ in our experiments.

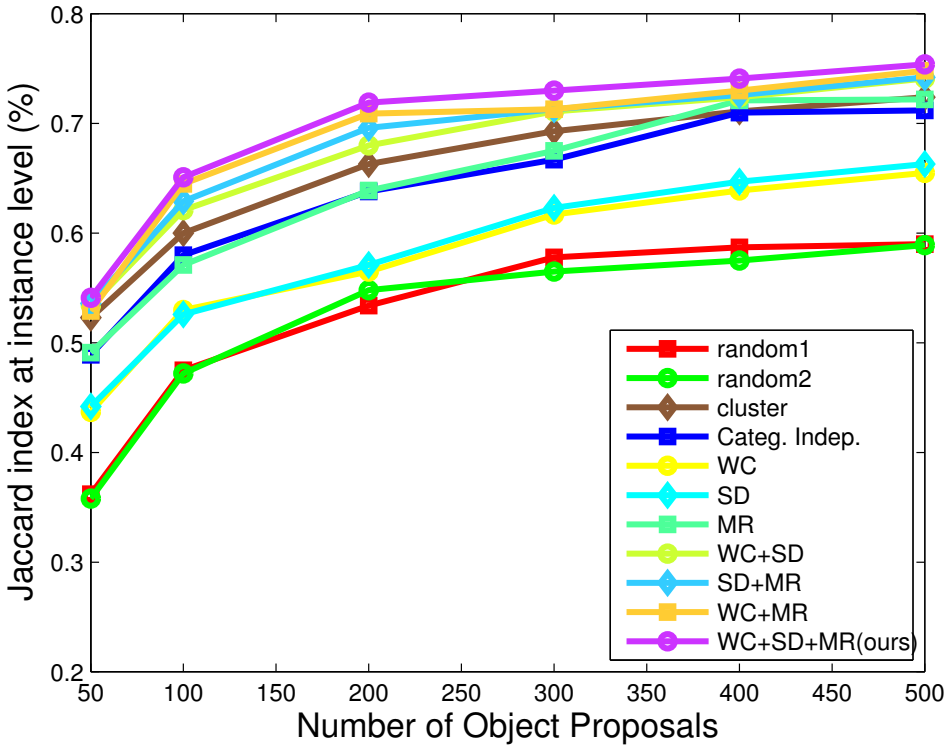


Figure 1.7: Comparing different ranking methods (random selection, clustering, Categ. Indep. [2], WC, SD, MR, WC+SD, WC+MR, SD+MR, WC+SD+MR(ours)).

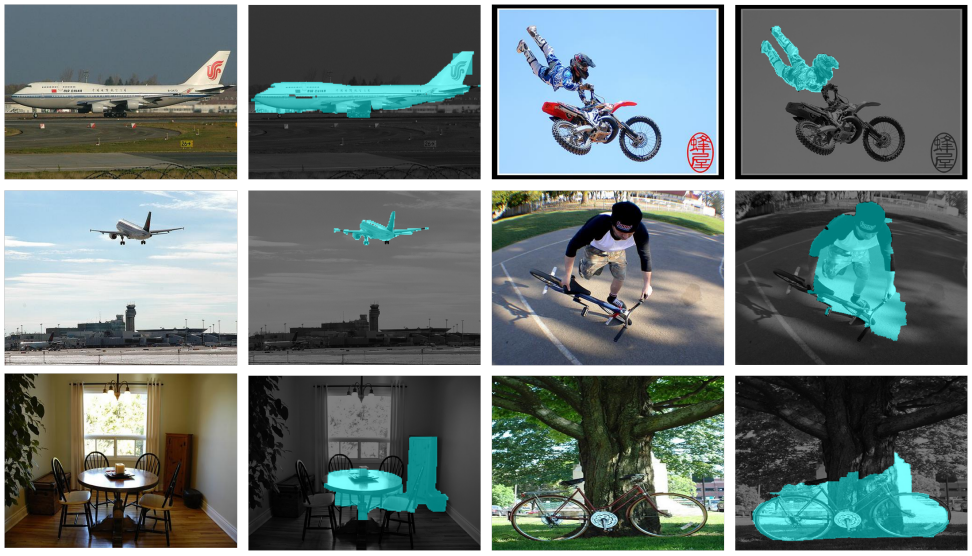


Figure 1.8: Top detections on: aeroplane, person, dining table, bicycle.Our detection results work well on objects of different scales.

1.6.2 Ranking performance

To explore our method's ranking ability, we compare our ranking method with four baselines on the PASCAL VOC2012 dataset. 1) **Random1** randomly selects object proposals from the multi-scale segment pool. 2) **Random2** randomly selects object proposals from each layer evenly, and combine them together. 3) **Clustering** selects the object proposals which are closest to the cluster center based on euclidean distance. The cluster centres are obtained via k-means clustering and k is set to be the number of object proposals to be selected. 4) **Categ. Indep.** is the method from [2] to rank segments. In order to show the importance of each term in our model, we evaluate each term: the weighted coverage term(WC), the single-layer diversity term (SD), and the multi-scale reward term (MR). Results of different term combinations (WC+SD, WC+MR, SD+MR)

and the full model (WC+SD+MR) are also presented.Figure 1.7 shows the quality of the selected object proposals using different ranking methods from the same segment pool.

1.6.3 Semantic Segmentation and Object Detection

To analyse the utility of the object proposals generated by our approach in real object recognition tasks, we perform semantic segmentation and object detection on the PASCAL VOC2012 validation set. We follow the settings in [22], where 2000 object proposals are generated for each image using our algorithm. Then we extract CNN features for both the regions and their bounding boxes using the deep convolutional neural network model pre-trained on ImageNet and fine-tuned on the PASCAL VOC2012 training set, the same as in [22]. These features are concatenated, then passed through linear classifiers trained for region and box classification tasks. After non-maxima suppression, we select the top 20,000 detections for each category.

The results are evaluated with the traditional bounding box AP^b and the extended metric AP^r as in [22] (the superscripts b and r correspond to region and bounding box). The AP^r score is the average precision of whether a hypothesis overlaps with the ground-truth instance by over 50%, and the AP^r_{vol} is the volume under the precision recall (PR) curve, which are suitable for the simultaneous segmentation and detection task. The evaluation of the detection task uses AP^b and AP^b_{vol}, which are conventional evaluation metric for object detection.

Table 1.3 and Table 1.4 shows the AP^r and AP^r_{vol} results for each class. We can see that the results using our object proposals, both our mean AP^r and mean AP^r_{vol} have

	RCNN	RCNN-MCG	SDS-A	Ours
mean AP^b	51.0	51.7	51.9	**52.4**
mean AP^b_{vol}	41.9	42.4	43.2	**44.3**

Table 1.5: Results on AP^b and AP^b_{vol} on the PASCAL VOC2012 val. All numbers are $\%$.

achieved state of the art using a seven-layer network, and we outperform previous methods in 14 out of 20 classes. In contrast to SDS [22], we neither fine tune different networks for regions and boxes nor refine the regions after classification. But our results still not only outperform the corresponding SDS-A but also the complicated SDS-B and SDS-C methods which finetuned two networks separately and as a whole. Moreover, on the more meaningful measurement of AP^r_{vol} shown in Table 1.4, results based on our object proposals even outperform that of SDS-C+ref, where the segments are refined within their 10×10 grid using a pretrained model with class priors. It shows the importance of good quality regions even before carefully designed feature extraction and region refinement after classification.

Table 1.5 shows the mean AP^b and mean AP^b_{vol} results for object detection. We achieved better results than RCNN [30], RCNN-MCG [22] and SDS-A [22], which shows that better region proposals not only improve segmentation but also give better localization of objects. Figure 1.8 shows some examples of our detection results.

1.7 Conclusion

We presented an efficient approach to extract multi-scale object proposals. Built on the top of hierarchical image segmentation, exemplar clusters are first generated within each scale to discover different object patterns. By introducing a weighted coverage term, a single-scale diversity term and a multi-scale reward term, we define a submodular objective function to select object proposals from multiple scales. The problem is solved using a highly efficient greedy algorithm with guaranteed performance. The experimental results on the BSDS dataset and the PASCAL VOC2012 dataset demonstrate that our method achieves state-of-art performance and is computationally efficient. We further evaluate our object proposals on a simultaneous detection and segmentation task to demonstrate the effectiveness of our approach and outperform the object proposals generated by other methods.

Chapter 2: Generating Object Proposals by Learning a Mixture of Submodular Functions

2.1 Overview

In this chapter, we extend the previous work using a supervised structured learning method to learn the importance of each term in the proposal ranking process. Structured learning has been used to solve multi-objective problems in several work. [37] introduced an algorithm to learn a mixture of submodular shells for document summarization and provided a risk bound guarantee. [38] focused on learning a mixture of sbumodular functions for video summarization. Each function in [38] measures interestingness, representativeness and uniformity of the video summary. We present novel submodular functions to measure representativeness, discriminativeness and multi-scale nature of object proposals. To our best knowledge, we are the first to use structure learning with submodular functions for object proposal generation.

2.2 Submodular Proposal Ranking

We start with a large pool of segments (or bounding boxes) from different image scales generated via hierarchical image segmentation. The task of object proposal ranking

is formulated as a subset selection problem. Given the large pool V, a limited number of segments are chosen as the object proposals. The problem is formulated as:

$$A = \arg max_{A \subset V} F(A), \quad |A| \leq K \tag{2.1}$$

where A is the selected subset of segments, K is the maximum number of segments to be chosen, and $F(A)$ is a linear combination of submodular terms with non-negative coefficients, defined as:

$$F(A) = \mathbf{w} \times \mathbf{f}(\mathbf{A}), \quad \mathbf{w} > \mathbf{0} \tag{2.2}$$

where $\mathbf{f}(A) = [f^1(A), f^2(A), ..., f^n(A)]^T$. Each f^i designs a submodular score function to measure one aspect of the selected subset.

2.2.1 Structured Learning

We want to minimize the chance of choosing inappropriate segments by minimizing a score function of submodular terms. We follow the maximum margin algorithm [37] to learn the weight vector $\mathbf{w}$ of Equation (A.4) so that the score of a good quality subset A_g has a higher score than all other subset $A \subset V$. The learning problem is formulated as follow:

$$\min_{\mathbf{w}>\mathbf{0}} \frac{1}{T} \sum_{i=1}^{N} L_i(\mathbf{w}) + \frac{\lambda}{2}||\mathbf{w}||^2 \tag{2.3}$$

where the subscript i denotes the features and subsets of segments in image i. The generalized hinge loss L_i is defined as:

$$L_i(\mathbf{w}) \triangleq \max_{A \subset V} (\mathbf{w}^T \mathbf{f_i}(A) + l_t(A)) - \mathbf{w}^T \mathbf{f_i}(A_g) \tag{2.4}$$

We adopt a recall loss $l_t(A)$ similar to the one in [38], $l_t(A) = (|A| - |A \bigcap A_g|)/K$. Since each image only contains a few number of ground truth objects, we compute the intersection area of the segment from multi-scale segment pool with the ground truth and rank them accordingly. The top K segments are used to compose the good quality set A_g in the training process. In this task, we define $\mathbf{f_i}(\mathbf{A}) = [f_i^{wc}(A), f_i^{sd}(A), f_i^{mr}(A)]^T$ (see detail in 2.2.2). As proved in [37], using approximate inference on learning submodular mixture have guaranteed performance using the existing subgradient descent algorithm.

2.2.2 Submodular Terms

The submodular terms are the weighted coverage term, single-scale diversity term, and multi-scale reward term defined the same as in Chapter 1.

2.2.2.1 Weighted Coverage Term

The weighted coverage term measures the representativeness of the selected subset A to the whole set V. Compared to generating object proposals from the raw superpixels, ranking pre-extracted segments from different scales is to select from a high-quality group. The selected subset A should be representative of the whole set V. Accordingly, we introduce the weighted coverage term that favours representative segments.

The weighed coverage term is formulated as:

$$f^{wc}(A) = \sum_{i \in V} \max_{j \in A} w_{ij} \qquad (2.5)$$
$$s.t. \quad A \subseteq V, |A| \leqslant K$$

where K is the maximum number of segments in the subset A. $\max_{j \in A} w_{ij}$ is the weighted

coverage of each segment v_i. Equation (2.5) measures the representativeness of A to V and favours segments that can cover (or represent) the other unselected segments well.

2.2.2.2 Single-Scale Diversity Term

The single-scale diversity term models the diversity among segments within each scale. The weighted coverage term will give rise to a highly representative set A; however, segments from each layer (corresponding to each image scale) possess redundancy. Therefore, we introduce a diversity term to encourage discriminativeness among segment within a layer. The single-layer diversity term is formulated as follows:

$$f^{sd}(A) = \sum_{l=1}^{L} D_l(A) = \sum_{t,l} \sqrt{\sum_{j \in P_t^l \cap A} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} \tag{2.6}$$

where P_t^l is the set of segments which belong to cluster t in layer l and V^l is the set of segments from layer l. The clusters within each layer is generated using kmeans algorithm with segment center location information. $|V^l|$ is the number of segments in layer l. This single-scale diversity shell encourage the diversity in the subset A by diminishing the benefit of choosing segments from the same cluster.

2.2.2.3 Multi-Scale Reward Term

The multi-scale reward term measures the likelihood of segments to be objects. As segments are from different scales, it is not appropriate to rank the rewards of all segments together. We propose the following discriminative multi-scale reward term to encourage

selected segments to have high likelihood to objects:

$$f^{mr}(A) = \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap A} r_j} \tag{2.7}$$

The value r_j estimates the likelihood of a segment to be an object. It determines the priority of a segment being chosen in its layer. We use segment features to train a SVM model over object segments and non-object segments in training images and then assign a confidence score for each segment during testing. The confidence score is used as r_j for a segment v_j.

2.2.3 Optimization

With the learnt weights from 2.2.1, we now generate object proposals using the objective function (A.4). The submodularity of $F(A)$ is preserved as it is a non-negative linear combination of three submodular terms $f^{wc}(A)$, $f^{sd}(A)$, and $f^{mr}(A)$. Direct maximization of equation (A.4) is an NP-hard problem. We can approximately solve the problem via a greedy algorithm [29, 32] based on its submodularity property. A lower bound of $(e-1)/e$ times the optimal value is guaranteed as proved in [29] (e is the base of the natural logarithm).

The algorithm starts from an empty set $A = \varnothing$. It adds the element a^* which provides the largest marginal gain among the unselected elements to A iteratively. The iterations stop when $|A|$ reaches the desired capacity number K. The optimization steps can be further accelerated using a lazy greedy approach from [23]. Instead of recomputing gain for every unselected element after each iteration, an ordered list of marginal benefits will be maintained in descending order. Only the top unselected segment is re-evaluated

at each iteration. Other unselected segments will be re-evaluated only if the top segment does not remain at the top after re-evaluation. The pseudo code is presented in Algorithm 2.

Algorithm 2 Submodular object proposal generation

Input: I, $G = (V, E)$, K, $\mathbf{w}$

Output: A

Initialization: $A \leftarrow \varnothing$, $U \leftarrow V$

loop

$a^* = arg\max_{a \in U} F(A \cup \{a\}) - F(A)$

if $|A| \geq K$ **then**

break

$A \leftarrow A \cup \{a^*\}$

$U \leftarrow U - \{a^*\}$

2.2.3.1 Structure Weights Learning

We demonstrate weight learning process in Figure 2.1. The weight is tuned on the training set of PASCAL2012 for selecting 2000 object proposals. The figure is shown on the first 1000 images. We can see it the importance of the individual submodular term converges quickly. The multi-scale reward term plays an import role in the object ranking process. This is reasonable for this dataset, where each image only contains a small amount of ground truth objects. Aiming at covering all objects, the diversity and representativeness of proposals which are captured by weighted coverage and single-scale diversity term should be relatively easy to satisfy with thousands of proposals.

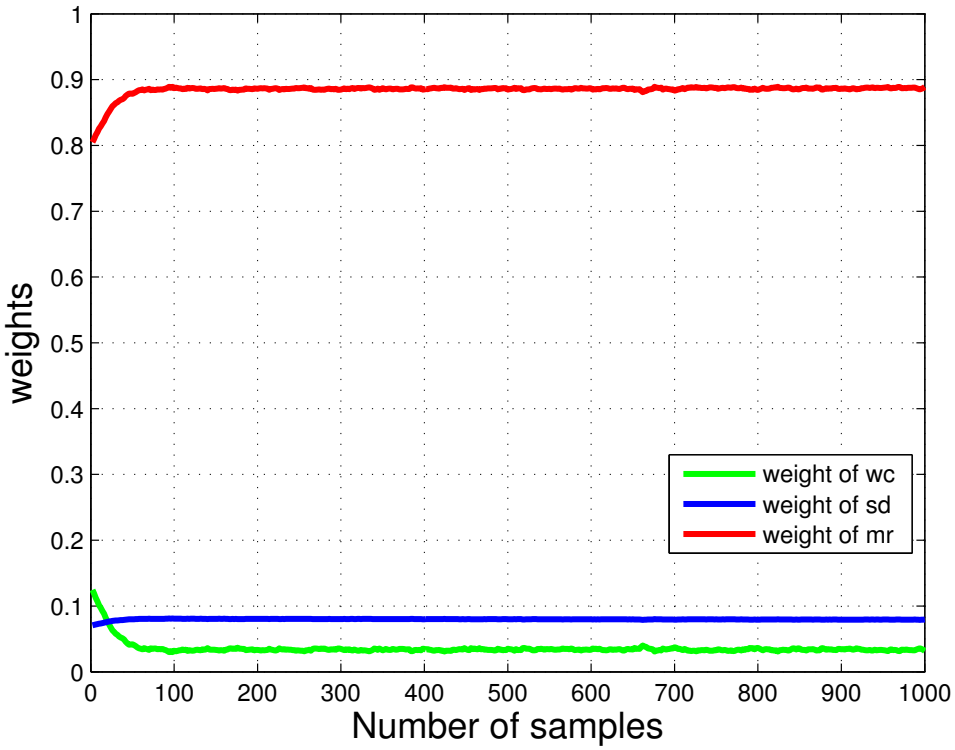


Figure 2.1: Weight learning. The importance of the weighted coverage (wc) term, the single-scale diversity (sd) term, and multi-scale reward (mr) term.

Chapter 3: Learning Structured Low-rank Representations for Image Classification

3.1 Overview

Image classification is one of the most fundamental task in computer vision. After obtaining an image, we would like the computer vision system to automatically decide what object is in the image. In this chapter, we propose better representations for image classification problem.

Recent research has demonstrated that sparse coding (or sparse representation) is a powerful image representation model. The idea is to represent an input signal as a linear combination of a few items from an over-complete dictionary D. It achieves impressive performance on image classification [39–41]. Dictionary quality is a critical factor for sparse representations. The sparse representation-based coding (SRC) algorithm [40] takes the entire training set as dictionary. However, sparse coding with a large dictionary is computationally expensive. Hence some approaches [40,42–44] focus on learning compact and discriminative dictionaries. The performance of algorithms like image classification is improved dramatically with a well-constructed dictionary and the encoding step is efficient with a compact dictionary. The performance of these methods deterio-

rates when the training data is contaminated (*i.e.*, occlusion, disguise, lighting variations, pixel corruption). Additionally, when the data to be analyzed is a set of images which are from the same class and sharing common (correlated) features (e.g. texture), sparse coding would still be performed for each input signal independently. This does not take advantage of any structural information in the set.

Low-rank matrix recovery, which determines a low-rank data matrix from corrupted input data, has been successfully applied to applications including salient object detection [45], segmentation and grouping [46–48], background subtraction [49], tracking [50], and 3D visual recovery [47, 51]. However, there is limited work [52, 53] using this technique for multi-class classification. [52] uses low-rank matrix recovery to remove noise from the training data class by class. This process becomes tedious as the class number grows, as in face recognition. Traditional PCA and SRC are then employed for face recognition. They simply use the whole training set as the dictionary, which is inefficient and not necessary for good recognition performance [54, 55]. [53] presents a discriminative low-rank dictionary learning for sparse representation (DLRD_SR) to learn a low-rank dictionary for sparse representation-based face recognition. A sub-dictionary D_i is learned for each class independently; these dictionaries are then combined to form a dictionary $D = [D_1, D_2, ...D_N]$ where N is the number of classes. Optimizing sub-dictionaries to be low-rank, however, might reduce diversity across items within each sub-dictionary. It results in a decrease of the dictionary's representation power.

We present a discriminative, structured low-rank framework for image classification. Label information from training data is incorporated into the dictionary learning process by adding an ideal-code regularization term to the objective function of dictio-

nary learning. Unlike [53], the dictionary learned by our approach has good reconstruction and discrimination capabilities. With this high-quality dictionary, we are able to learn a sparse and structural representation by adding a sparseness criteria into the low-rank objective function. Images within a class have a low-rank structure, and sparsity helps to identify an image's class label. Good recognition performance is achieved with only one simple multi-class classifier, rather than learning multiple classifiers for each pair of classes [43, 56, 57]. In contrast to the prior work [52, 53] on classification that performs low-rank recovery class by class during training, our method processes all training data simultaneously. Compared to other dictionary learning methods [40, 54, 55, 58] that are very sensitive to noise in training images, our dictionary learning algorithm is robust. Contaminated images can be recovered during our dictionary learning process.

3.1.1 Related Work

Sparse representation has been widely used for image classification. [59] has shown that sparse representation achieves impressive results on face recognition. The entire training set is taken as the dictionary. [39, 60] formulate a sparsity-constrained framework to model the sparse coding problem. They use a modified model to handle corruptions like occlusion in face recognition. These algorithms, however, don't learn a dictionary. The selection of the dictionary, as shown in [61], can strongly influence classification accuracy. One of the most commonly used dictionary learning method is K-SVD [42]. This algorithm focuses on the representation power of dictionaries. Several algorithms have been developed to make the dictionary more discriminative for sparse coding. In [44], a

dictionary is updated iteratively based on the results of a linear predictive classier to include structure information. [54] presents a Label Consistent K-SVD (LC-KSVD) algorithm to learn a compact and discriminative dictionary for sparse coding. These methods show that performance is improved dramatically with a structured dictionary. However, if the training data is corrupted by noise, their performance is diminished.

Using low-rank matrix recovery for denoising has attracted much attention recently. Wright introduced the Iterative Thresholding Approach [59] to solve a relaxed convex form of the problem. The Accelerated Proximal Gradient Approach is described in [59, 62]. The Dual Approach in [62] tackles the problem via its dual. Applying augmented Lagrange multipliers (ALM), Lin [63] proposed RPCA via the Exact and Inexact ALM Method. Promising results have been shown in many applications [45–48, 50]. Limited work, however, has applied the low-rank framework to solve image classification problems. [52] uses a low-rank technique to remove noise from training data. Denoising is implemented class by class, which gives rise to tremendous computational cost as class number increases. [53] enhances a sparse coding dictionary's discriminability by learning a low-rank sub-dictionary for each class. This process is time-consuming and might increase the redundancy in each sub-dictionary, thus not guaranteeing consistency of sparse codes for signals from the same class. [51] presents an image classification framework by using non-negative sparse coding, low-rank and sparse matrix decomposition. A linear SVM classifier is used for the final classification.

Compared to previous work, our approach effectively constructs a reconstructive and discriminative dictionary from corrupted training data. Based on this dictionary, structured low-rank and sparse representations are learned for classification.

3.2 Low-rank Matrix Recovery

Suppose a matrix X can be decomposed into two matrices, *i.e.*, $X = A + E$, where A is a low-rank matrix and E is a sparse matrix. Low-rank matrix recovery aims at finding A from X. It can be viewed as an optimization problem: decomposing the input X into $A + E$, minimizing the rank of A and reducing $||E||_0$.

$$\min_{A,E} rank(A) + \lambda||E||_0 \quad s.t\ X = A + E \tag{3.1}$$

where λ is a parameter that controls the weight of the noise matrix E. However, direct optimization of (3.1) is NP-hard. [64] shows that if the rank of A is not too large and E is sparse, the optimization problem is equivalent to:

$$\min_{A,E} ||A||_* + \lambda||E||_1 \quad s.t\ X = A + E \tag{3.2}$$

where $||A||_*$ is the nuclear norm (i.e., the sum of the singular values) of A. It approximates the rank of A. $||E||_0$ could be replaced with the l_1-norm $||E||_1$. As proved in [64], low-rank and sparse components are identifiable. Under fairly general conditions, A can be exactly recovered from X as long as E is sufficiently sparse (relative to the rank of A) [59]. This model assumes that all vectors in X are coming from a single subspace. [52] uses this technique to remove noise from training samples class by class; this process is computationally expensive for large numbers of classes. Moreover, structure information is not well preserved. [52] solves this problem by promoting the incoherence between different classes. A regularization term $\eta \sum_{j \neq i} ||A_j^T A_i||_F^2$ is added to function (3.2). It needs to be updated whenever A_j is changed. This is complicated and might not be helpful for classification.

Consider the problem of face recognition. Here, the dataset is a union of many subjects; samples of one subject tend to be drawn from the same subspace, while samples of different subjects are drawn from different subspaces. [65] proves that there is a lowest-rank representation that reveals the membership of samples. A more general rank minimization problem [65] is formulated as:

$$\min_{Z,E} ||Z||_* + \lambda||E||_{2,1} \tag{3.3}$$
$$s.t\ X = DZ + E$$

where D is a dictionary that linearly spans the data space. The quality of D will influence the discriminativeness of the representation Z. [65] employs the whole training set as the dictionary, but this might not be efficient for finding a discriminative representation in classification problems. [53] tries to learn a structured dictionary by minimizing the rank of each sub-dictionary. However, it reduces diversity in each sub-dictionary, thus weakening the dictionary's representation power.

We will show that an efficient representation can be obtained with respect to a well-structured dictionary. Associating label information in the training process, a discriminative dictionary can be learned from all training samples simultaneously. The learned dictionary encourages images from the same class to have similar representations (*i.e.*, lie in a low-dimensional subspace); while images from other classes have very different representations. This leads to high recognition performance of our approach, as shown in the experiment section.

3.3 Learning Structured Sparse and Low-rank Representation

To better classify images even when training and testing images have been corrupted, we propose a robust supervised algorithm to learn a structured sparse and low-rank representation for images. We construct a discriminative dictionary via explicit utilization of label information from the training data. Based on the dictionary, we learn low-rank and sparse representations for images. Classification is carried out directly on these discriminative representations.

3.3.1 Problem Statement

We are given a data matrix $X = [X_1, X_2, ..., X_N]$ with N classes where X_i corresponds to class i. X may be contaminated by noise (occlusion, corruption, illumination differences, etc). After eliminating noise, samples within each class i will demonstrate similar basic structure [65, 66]. As discussed before, low-rank matrix recovery helps to decompose a corrupted matrix X into a low-rank component DZ and a sparse noise component E, *i.e.*, $X = DZ + E$. With respect to a semantic dictionary D, the optimal representation matrix Z for X should be block-diagonal [65]:

$$Z^* \triangleq \begin{pmatrix} Z_1^* & 0 & 0 & 0 \\ 0 & Z_2^* & 0 & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & Z_N^* \end{pmatrix}$$

Based on the above discussion, it is possible to learn low-rank and sparse representations for images. Low rankness reveals structure information. Sparsity identifies which

class an image belongs to. Given a dictionary D, the objective function is formulated as:

$$\min_{Z,E} ||Z||_* + \lambda||E||_1 + \beta||Z||_1 \tag{3.4}$$

$$s.t \quad X = DZ + E$$

where λ, β controls the sparsities of the noise matrix E and the representation matrix Z, respectively.$||.||_*$ and $||.||_1$ denotes the nuclear norm and the l_1-norm of a matrix.

The dictionary $D = [D_1, D_2, ...D_N]$ contains N sub-dictionaries where D_i corresponds to class i. Let $Z_i = [Z_{i,1}, Z_{i,2}, ...Z_{i,N}]$ be the representation for X_i with respect to D. Then $Z_{i,j}$ denotes coefficients for D_j. To obtain a low-rank and sparse data representation, D should have discriminative and reconstructive power. Firstly, D_i should ideally be exclusive to each subject i. Thus, representations for images from different classes would be different. Secondly, every class i is well represented by its sub-dictionary such that $X_i = D_i Z_{i,i} + E_i$. $Z_{i,j}$, the coefficients for D_j $(i \neq j)$, are nearly all zero.

We say Q is an ideal representation if $Q = [q_1, q_2, ..., q_T] \in R^{K \times T}$ where q_i, the code for sample x_i, is of the form of $[0...1, 1, 1, ...]^t \in R^K$ (K is the dictionary's size, and T is the total number of samples). Suppose x_i belongs to class L, then the coefficients in q_i for D_L are all 1s, while the others are all 0s. An example optimal decomposition for image classification is illustrated in Figure 3.1. Here, data $X = [X_1, X_2, X_3]$ contains images from 3 classes, where X_1 contains 3 samples x_1, x_2, x_3, X_2 contains 4 samples x_4, x_5, x_6, x_7, and X_3 contains 3 samples x_8, x_9, x_{10}. D has 3 sub-dictionaries, and each has 2 items. Although this decomposition might not result in minimal reconstruction error, low-rank and sparse Q is an optimal representation for classification.

With the above definition, we propose to learn a semantic structured dictionary by

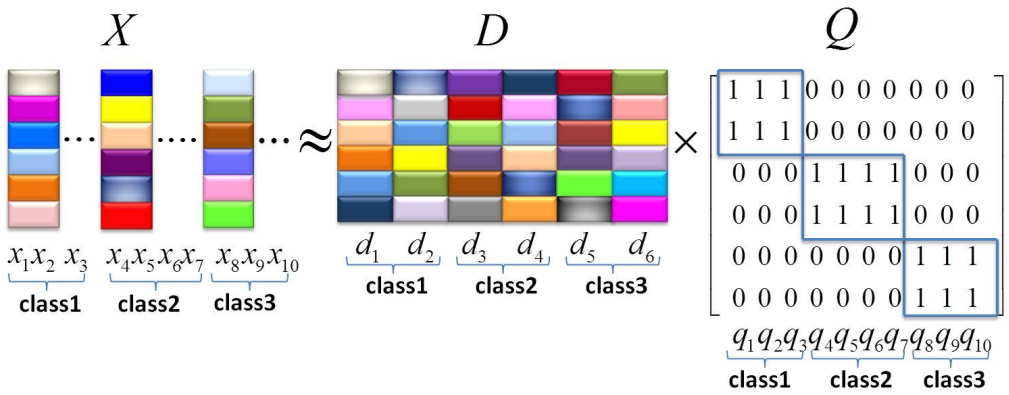


Figure 3.1: Optimal decomposition for classification.

supervised learning. Based on label information, we construct Q in block-diagonal form for training data. We add a regularization term $||Z-Q||_F^2$ to include structure information into the dictionary learning process. A dictionary that encourages Z to be close to Q is preferred. The objective function for dictionary learning is defined as follows:

$$\min_{Z,E,D} ||Z||_* + \lambda||E||_1 + \beta||Z||_1 + \alpha||Z-Q||_F^2 \tag{3.5}$$
$$s.t \quad X = DZ + E$$

where α controls the relative contribution of the regularization term.

3.3.2 Optimization

To solve optimization problem (3.5), we first introduce an auxiliary variable W to make the objective function separable. Problem (3.5) can be rewritten as:

$$\min_{Z,E,D} ||Z||_* + \lambda||E||_1 + \beta||W||_1 + \alpha||W-Q||_F^2 \tag{3.6}$$
$$s.t \quad X = DZ + E, W = Z$$

The augmented Lagrangian function L of (3.6) is:

$$\begin{aligned} & L(Z, W, E, D, Y_1, Y_2, \mu) \\ = \; & ||Z||_* + \lambda||E||_1 + \beta||W||_1 + \alpha||W - Q||_F^2 \\ & + < Y_1, X - DZ - E > + < Y_2, Z - W > \\ & + \frac{\mu}{2}(||X - DZ - E||_F^2 + ||Z - W||_F^2) \end{aligned} \tag{3.7}$$

where $< A, B >= trace(A^t B)$.

The optimization for problem (3.6) can be divided into two subproblems. The first subproblem is to compute the optimal Z, E for a given dictionary D. If we set $\alpha = 0$, this is exactly the optimization problem from (3.4). The second subproblem is to solve dictionary D for the given Z, E calculated from the first subproblem.

3.3.2.1 Computing Representation Z Given D

With the current D, we use the linearized alternating direction method with adaptive penalty (LADMAP) [67, 68] to solve for Z and E. The augmented Lagrangian function (3.7) can be rewritten as:

$$\begin{aligned} & L(Z, W, E, D, Y_1, Y_2, \mu) \\ = \; & ||Z||_* + \lambda||E||_1 + \beta||W||_1 + \alpha||W - Q||_F^2 \\ & + h(Z, W, E, D, Y_1, Y_2, \mu) - \frac{1}{2\mu}(||Y_1||_F^2 + ||Y_2||_F^2) \end{aligned} \tag{3.8}$$

where $h(Z, W, E, D, Y_1, Y_2, \mu)$

$= \frac{\mu}{2}(||X - DZ - E + \frac{Y_1}{\mu}||_F^2 + ||Z - W + \frac{Y_2}{\mu}||_F^2)$

The quadratic term h is replaced with its first order approximation at the previous iteration step adding a proximal term [67]. The function is minimized by updating each of the

variables Z, W, E one at a time. The scheme is as follows:

$$\begin{aligned} Z^{j+1} &= \arg\min_{Z} ||Z||_* + < Y_1^j, X - DZ^j - E^j > \\ &\quad + < Y_2^j, Z^j - W^j > + \frac{\mu}{2}(||X - D^j Z^j \\ &\quad - E^j||_F^2 + ||Z^j - W^j||_F^2) \\ &= \arg\min_{Z} ||Z||_* + \frac{\eta\mu}{2}||Z - Z^j||_F^2 \\ &\quad + < \nabla_Z h(Z^j, W^j, E^j, Y_1^j, Y_2^j, \mu), Z - Z^j > \\ &= \arg\min_{Z} \frac{1}{\eta\mu}||Z||_* + \frac{1}{2}||Z - Z^j + \big[- D^T(X - \\ &\quad DZ^j - E^j + \frac{Y_1^j}{\mu}) + (Z - W^j + \frac{Y_2^j}{\mu})\big]/\eta||_F^2 \end{aligned} \tag{3.9}$$

$$\begin{aligned} W^{j+1} &= \arg\min_{W} \beta||W||_1 + \alpha||W - Q||_F^2 \\ &\quad + < Y_2^j, Z - W > + \frac{\mu}{2}||Z^{j+1} - W||_F^2 \\ &= \arg\min_{W} \frac{\beta}{2\alpha + \mu}||W||_1 + \frac{1}{2}||W - (\frac{2\alpha}{2\alpha + \mu}Q \\ &\quad + \frac{1}{2\alpha + \mu}Y_2^j + \frac{\mu}{2\alpha + \mu}Z^{j+1})||_F^2 \end{aligned} \tag{3.10}$$

$$\begin{aligned} E^{j+1} &= \arg\min_{E} \lambda||E||_1 + < Y_1^j, X - DZ^{j+1} - E > \\ &\quad + \frac{\mu}{2}||X - DZ^{j+1} - E||_F^2 \\ &= \arg\min_{E} \frac{\lambda}{\mu}||E||_1 + \frac{1}{2}||E - (\frac{1}{\mu}Y_1^j + X \\ &\quad - DZ^{j+1})||_F^2 \end{aligned} \tag{3.11}$$

where $\nabla_Z h$ is the partial differential of h with respect to Z. $\eta = ||D||_2^2$. The calculations are described in Algorithm 3.

Algorithm 3 Low-Rank Sparse Representation via Inexact ALM

Input: Data X, Dictionary D, and Parameters λ, β, α

Output: Z, W, E

Initialize: $Z^0 = W^0 = E^0 = Y_1^0 = Y_2^0 = 0, \rho = 1.1, \epsilon = 10^{-7}, \mu_{max} = 10^{30}$

while not converged, $j \leq maxIterZ$ **do**

fix W, E and update variable Z according to (3.9)

fix Z, E and update variable W according to (3.10)

fix Z, W and update variable E according to (3.11)

update the multipliers:

$$Y_1^{j+1} = Y_1^j + \mu(X - DZ^j - E^j)$$

$$Y_2^{j+1} = Y_2^j + \mu(Z^j - W^j)$$

update μ:

$$\mu = \min(\mu_{max}, \rho\mu)$$

check the convergence conditions:

$$||X - DZ^j - E^j||_\infty < \epsilon, ||Z^j - W^j||_\infty < \epsilon$$

3.3.2.2 Updating Dictionary D with Fixed Z, W, E

With fixed Z, W and E, D is the only variable in this subproblem. So (3.7) can be rewritten as:

$$\begin{aligned} & L(Z, W, E, D, Y_1, Y_2, \mu) \\ = \; & < Y_1, X - DZ - E > + \frac{\mu}{2}(||X - DZ - E||_F^2 \\ & + ||Z - W||_F^2) + C(Z, E, W, Q) \end{aligned} \tag{3.12}$$

where $C(Z, E, W, Q)$ is fixed. This equation (3.12) is a quadratic form in variable D, so we can derive an optimal dictionary D^{update} immediately. The updating scheme is:

$$D^{i+1} = \gamma D^i + (1 - \gamma) D^{update} \tag{3.13}$$

γ is a parameter that controls the updating step. The dictionary construction process is summarized in Algorithm 4.

Algorithm 4 Dictionary Learning via Inexact ALM

Input: Data X, and Parameters λ, β, α, γ

Output: D, Z

Initialize: Initial Dictionary D^0, $\epsilon_d = 10^{-5}$

while not converged, $i \leq maxIterD$ **do**

find Z, W, E with respect to D^i using Algorithm 3

fix Z, W, E and update D by:

$D^{update} = \frac{1}{\mu}(Y_1 + \mu(X - E))Z^T(ZZ^T)^{-1}$

$D^{i+1} = \gamma D^i + (1 - \gamma) D^{update}$

check the convergence conditions:

$||D^{i+1} - D^i||_\infty < \epsilon_d$

3.3.2.3 Dictionary Initialization

To initialize the dictionary, we use the K-SVD method. The initial sub-dictionary D_i for class i is obtained by several iterations within each training class. The input dictionary D^0 is initialized by combining all the individual class dictionaries, *i.e.*, $D^0 = [D_1, D_2, ...D_N]$.

3.3.3 Classification

We use a linear classifier for classification. After the dictionary is learned, the low-rank sparse representations Z of training data X and Z_{test} of test data X_{test} are calculated by solving (3.4) separately using Algorithm 3 with $\alpha = 0$. The representation z_i for test sample i is the ith column vector in Z_{test}. We use the multivariate ridge regression model [69, 70] to obtain a linear classifier $\hat{W}$:

$$\hat{W} = \arg\min_{W} ||H - WZ||_2^2 + \lambda||W||_2^2 \tag{3.14}$$

where H is the class label matrix of X. This yields $\hat{W} = HZ^T(ZZ^T + \lambda I)^{-1}$. Then label for sample i is given by:

$$k = arg\max_{k}(s = \hat{W}z_i) \tag{3.15}$$

where s is the class label vector.

3.4 Experiments

We evaluate our algorithm on three datasets. Two face databases: Extended YaleB [71], AR [72], and one object category database: Caltech101 [73]. Our approach is compared

with several other algorithms including the locality-constrained linear coding method (LLC) [58], SRC [40], LR [52], LR with structural incoherence from [52], DLRD_SR [53] and our method without the regularization term $||Z - Q||$ (our method without Q). Our method without Q involves simply setting $\alpha = 0$ in the dictionary learning process. Unlike most other image classification methods [39,42,44], training and testing data can both be corrupted. Our algorithm achieves state of the art performance on various experiments.

3.4.1 Extended YaleB Database

The Extended YaleB database contains 2,414 frontal-face images of 38 people. Taken under various controlled lighting conditions, these cropped images have size 192×168 pixels. There are between 59 and 64 images for each person. Shadows due to different illumination conditions cause variations in this dataset. We test our algorithm on the original images as well as down-sampled images $(2, 4, 8)$. This results in data sets of feature dimension 32256, 8064, 2016 and 504. We randomly select 8 training images for each person, repeat this 5 times and report average recognition accuracy. Our trained dictionary has 5 items for each class. Then we repeat our experiments starting with 32 randomly selected training images and 20 dictionary items per class.

We compare our approach with LLC [58], SRC [40], LR [52], and LR with structural incoherence [52]. We evaluate the performance of the SRC algorithm using a full-size dictionary (all training samples). For fair comparison, we also evaluate the results of SRC and LLC using dictionaries whose sizes are the same with ours. The result for our method without Q is also calculated. The comparative results are shown in Figure 4.3.

n is the number of training samples for each person. Our method, by taking advantage of structure information, achieves better performance than LLC, LR, LR with structural incoherence and our method without Q. It outperforms SRC when using the same-size dictionary.

Figure 3.3 illustrates the representations for the first ten subjects. The dictionary contains 50 items (5 for each category). The first line shows the testing images' representation based on LR and LR with structural incoherence [52]. Figures 3.3(a) and 3.3(c) are representations with the full size dictionary (all training sample). For comparison, we randomly select 5 out of 8 training samples from each class, and generate a 50-element dictionary. The corresponding representations are shown in Figures 3.3(b) and 3.3(d). Figures 3.3(e), 3.3(f) and 3.3(g) are the representations based on SRC, LLC with the same dictionary size and our method without Q. In our learned representation, Figure 3.3(h), images from the same class show strong similarities. This representation is much more discriminative than the others.

We present some examples of decomposition results in Figure 3.4. The first three images are original faces. The middle and the last three images are the low-rank component (DZ) and the noise component (E), respectively. We see that different illumination conditions mainly influence the noise component.

We also evaluate the computation time of our approach and LR with structural incoherence [52] that trains a model class by class (Figure 3.5(a)) and uses SRC for classification. The training time is computed as the average over the entire training set. The testing time, which includes both encoding and classification, is averaged over all test samples. Clearly, training over all classes simultaneously is faster than class by class if

Table 3.1: Recognition rates on the AR

Dimension2200	sunglass	scarf	mixed
Our Method	87.3	83.4	82.4
Our Method without Q	85.1	81.3	81.0
LR w. Struct. Incoh. [52]	84.9	76.4	80.3
LR [52]	83.2	75.8	78.9
SRC(all train. samp.) [40]	86.8	83.2	79.2
SRC*(5 per person) [40]	82.1	72.6	65.5
LLC [58]	65.3	59.2	59.9

discriminativeness is preserved for different classes. Our training time is twice as fast and testing is three times faster than LR with structural incoherence.

3.4.2 AR Database

The AR face database includes over 4,000 color face images of 126 individuals, 26 images for each person in two sessions. In each session, each person has 13 images. Among them, 3 are obscured by scarves, 6 by sunglasses, and the remaining faces are of different facial expressions or illumination variations which we refer to as unobscured images. Each image is 165×120 pixels. We convert the color images into gray scale and down-sample 3×3. Following the protocol in [52], experiments are run under three different scenarios:

Sunglasses: In this scenario, we consider unobscured images and those with sunglasses. We use seven unobscured images from session 1 and one image with sunglass as

training samples for each person, the rest as testing. Sunglasses cover about 20% of the face.

Scarf: In this scenario, we consider unobscured images and those with scarves. We use seven unobscured images from session 1 and one image with a scarf as training samples for each person, the remainder as testing. Scarves give rise to around 40% occlusion.

Mixed (Sunglass + Scarf): In the last scenario, we consider all images together (sunglass, scarf and the unobscured). We use seven unobscured images from session 1, one image with sunglasses, and one with a scarf as training samples for each person.

We repeat our experiments three times for each scenario and average the results. Table 4.1 summarizes the results. We use $\alpha = 560$, $\lambda = 16$, $\beta = 15$, $\gamma = 0.1$ in our experiments. Our methods are compared with LLC [58], SRC [40], LR [52], and LR with structural incoherence [52]. For SRC, we measure the performance with two different dictionary sizes. Our approach achieves the best results and outperforms other approaches with the same dictionary size by more than 3% for the sunglass scenario, 7% for the scarf scenario, and 2% for the mixed scenario.

We visualize the representation Z for the first ten classes under the sunglasses scenario. There are $8 \times 10 = 80$ training images and $12 \times 10 = 120$ testing images. We use 50 as our dictionary size, *i.e.*, 5 dictionary items per class. Figures 3.6(a) and 3.6(c) show the representations of LR and LR method without structural incoherence with a full-size dictionary. In Figures 3.6(b) and 3.6(d), we randomly pick 5 dictionary items for each class, and use this reduced dictionary to learn sparse codes. For comparison purposes, we also choose 50 as the dictionary size in LLC and SRC* to learn the representations shown

in Figures 3.6(e) and 3.6(f). The testing images automatically generate a block diagonal structure in our method, which is absent in other methods.

Figure 3.7 shows image decomposition examples on the AR database. The first row shows the original gray images. The second is the low-rank component (DZ) and the third the noise component (E). Our approach separates occlusions such as sunglasses and scarves from the original images into the noise component.

Table 3.2: Recognition rates on the AR

Dimension2200	sunglass	scarf
Our Method	90.9	88.5
LC-KSVD [54]	78.4	63.7

In addition, we compare our results with LC-KSVD [54] using the same training samples under the sun and scarf scenarios, using unobscured images for test. The results is summarized in Table 4.2. Although associating label information with the training process, the performance of LC-KSVD is not as good as ours since the training set is smaller and corrupted. Our approach, in contrast, is robust to noise like occlusion.

We also evaluate our algorithm on the corrupted AR face database following the protocol in [53]. In the experiment, seven images with illumination and expression variations from session 1 are used for training images, and the other seven images from session 2 are used as testing images. A percentage of randomly chosen pixels from each training and testing image are replaced with iid samples from a uniform distribution over $[0, V_{max}]$ as [59] did, where V_{max} is the largest possible pixel value in the image. The recognition rates under different levels of noises are shown in Figure 3.5(b). The results

Table 3.3: Recognition rates on the Caltech101

number of training sample	15	30
Our Method	66.1	73.6
Our Method without Q	65.5	73.3
LR w. Struct. Incoh. [52]	58.3	65.7
LR [52]	50.3	60.1
SRC (all train. samp.) [40]	64.9	70.7
LLC [58]	65.4	73.4

of DLRD_SR [53], FDDL [60], Robust PCA [59], SR [40], and SVM [59] are copied from [53]. Our method outperforms the other approaches. Figure 3.8 shows some examples of image decomposition on the AR database with 20% uniform noise.

3.4.3 Caltech101 Database

The Caltech101 database contains over 9000 images from 102 classes. 101 classes are of animals, flowers, trees, etc. and there is a background class. The number of images in each class is between 31 and 800. We evaluate our methods using spatial pyramid features and run experiments with 15 and 30 randomly chosen training images.

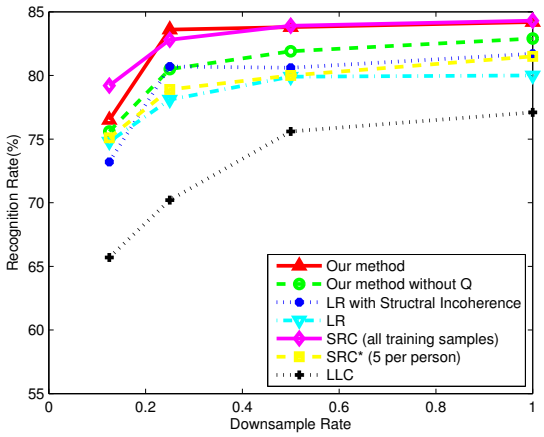
Figure 3.9 shows the representations of 15 testing samples which are randomly selected from classes $4 \sim 8$. Our representation clearly reveals structure information through representation similarity. Although the training images are visually very diverse, we are able to learn discriminative representations with the constructed dictionary.

We evaluate our approach and compare it with others [40,52,58]. Table 3.3 presents

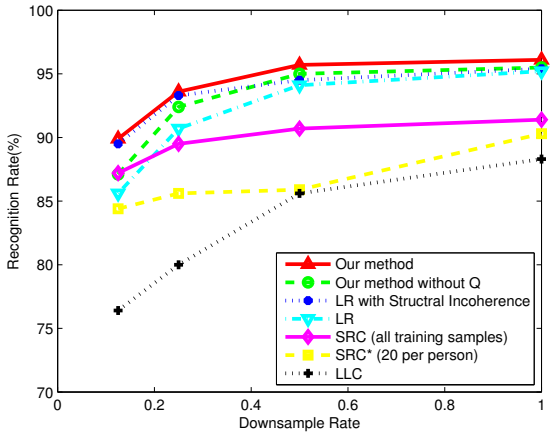
classification accuracy. Our algorithm achieves the best performance. Figure 3.10 gives some examples from the classes which achieve high classification accuracy when the number of training images is 30 per category.

3.5 Conclusions

We proposed a new image classification model to learn a structured low-rank representation. Incorporating label information into the training process, we construct a semantic structured and constructive dictionary. Discriminative representations are learned via low-rank recovery even for corrupted datasets. The learned representations reveal structural information automatically and can be used for classification directly. Experiments show our approach is robust, achieving state-of-art performance in the presence of various sources of data contamination, including illumination changes, occlusion and pixel corruption.

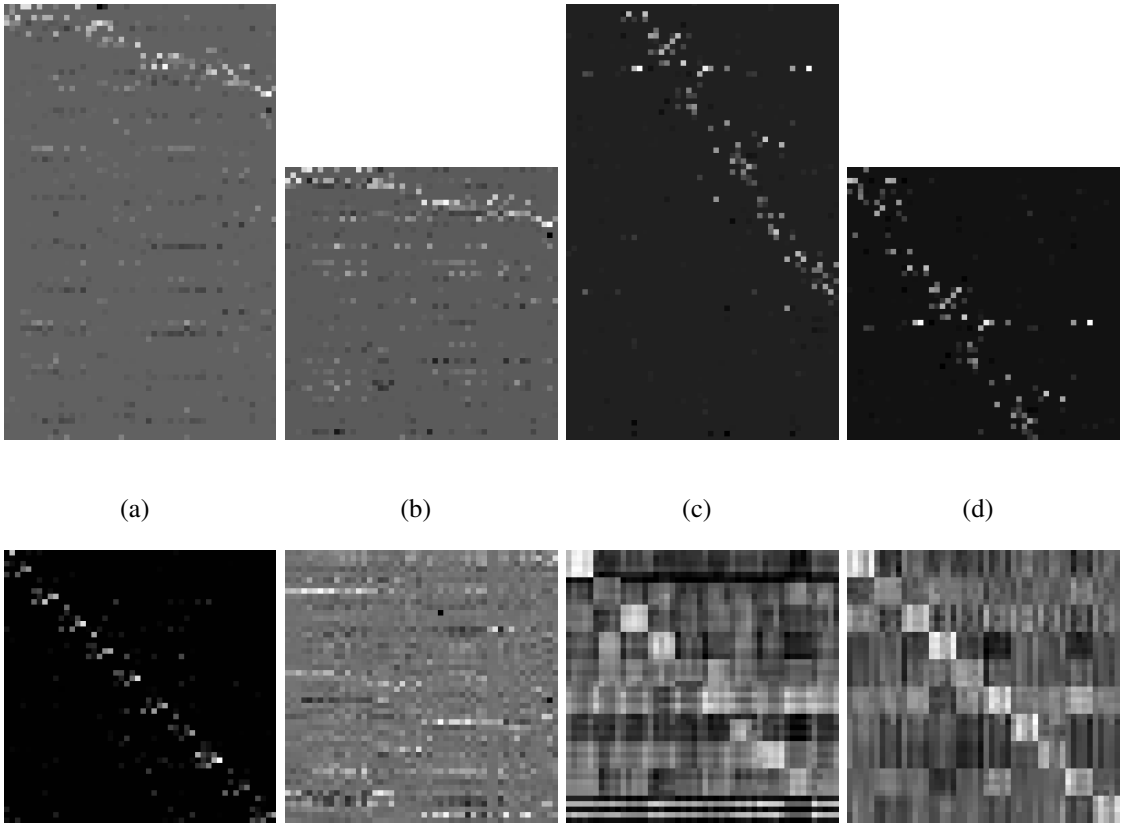


(a) n = 8



(b) n = 32

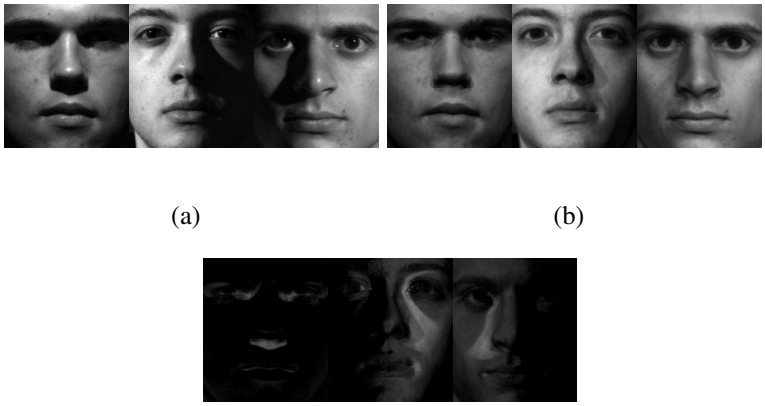
Figure 3.2: Performance comparisons on the Extended YaleB. n is the number of training images per person.



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

(e) (f) (g) (h)

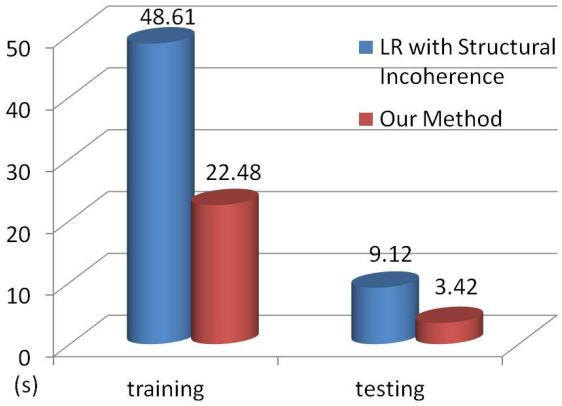
Figure 3.3: Comparison of representations for testing samples from the first ten classes on the Extended YaleB. 5 example samples for each class. (a) LR with full-size dictionary; (b) LR with dictionary size 50; (c) LR with structural incoherence with full-size dictionary; (d) LR with structural incoherence with dictionary size 50; (e) SRC; (f) LLC; (g) Our method without Q; (h) Our method.



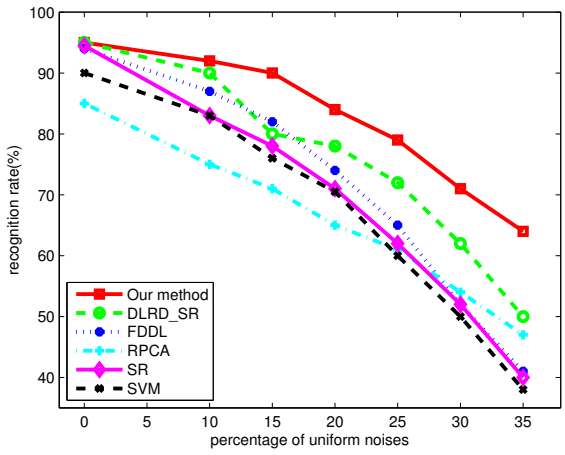
(a) (b)

(c)

Figure 3.4: Examples of image decomposition for testing samples on the Extended YaleB. (a) original faces; (b) the low-rank component DZ; (c) the sparse noise component E.

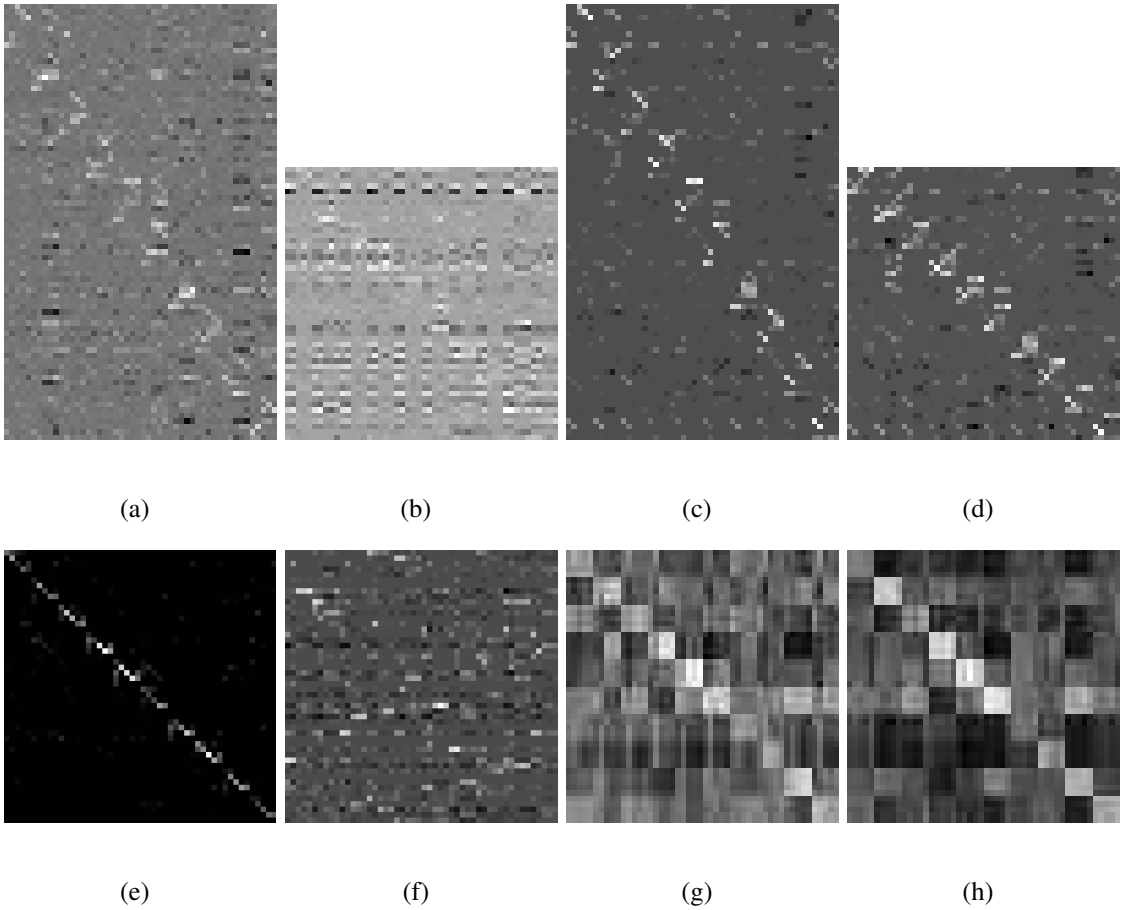


(a)



(b)

Figure 3.5: Experiment results. (a) Average computation time for training and testing on the Extended YaleB; (b) Recognition rates on the AR database with pixel corruption.



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

(e) (f) (g) (h)

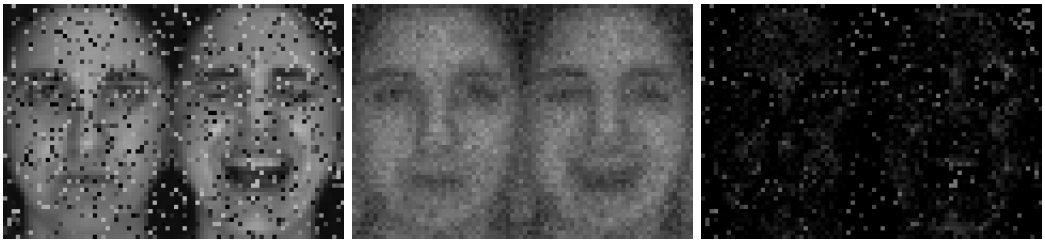
Figure 3.6: Comparison of representations for testing samples from the first ten classes on the AR for the sunglass scenario. 5 samples for each class. (a) LR with full-size dictionary; (b) LR with dictionary size 50; (c) LR with structural incoherence with full-size dictionary; (d) LR with structural incoherence with dictionary size 50; (e) SRC; (f) LLC; (g) Our method without Q; (h) Our method.

(a) original gray images

(b) the low-rank component DZ

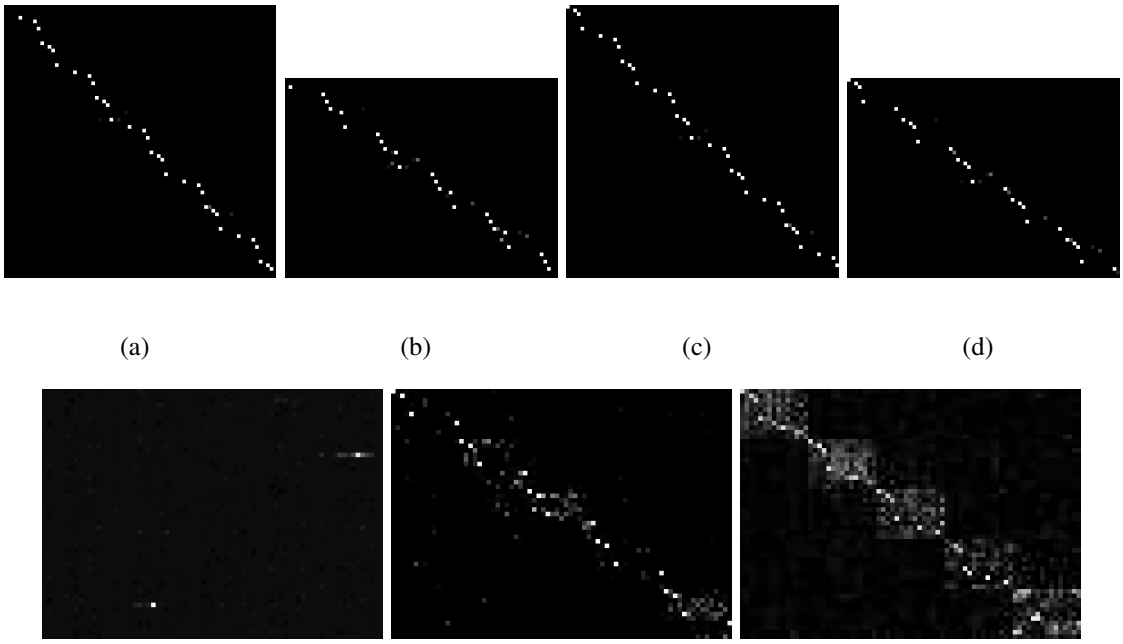
(c) the sparse noise component E

Figure 3.7: Examples of image decomposition for testing samples from class 4 and 10 on the AR.



(a) (b) (c)

Figure 3.8: Examples of image decomposition for testing samples from class 95 on the AR with 20% uniform noise. (a) corrupted faces; (b) the low-rank component DZ; (c) the sparse noise component E.



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

(e) (f) (g)

Figure 3.9: Comparison of representations for testing samples from class 4 to 8 on the Caltech101. 15 example samples for each class. (a) LR with full-size dictionary; (b) LR with dictionary size 55; (c) LR with structural incoherence with full-size dictionary; (d) LR with structural incoherence with dictionary size 55; (e) LLC; (f) Our method without Q; (g) Our method.

(a) yin_yang, acc:100%

(b) soccer_ball, acc:100%

(c) sunflower, acc:100%

(d) Motorbikes, acc:97.7%

(e) accordion, acc:96.0%

(f) watch, acc:95.7%

Figure 3.10: Example images from classes with high classification accuracy of the Caltech101.

Chapter 4: Discriminative Tensor Sparse Coding for Image Classification

4.1 Overview

To extend the dictionary learning to multi-dimensional feature, we present a discriminative tensor sparse coding model for image classification problem. Sparse models have been successfully applied to many problems in image processing, computer vision, and machine learning. Many algorithms [54, 74] have been proposed to learn an overcomplete and compact dictionary based on such models. In general, the input feature representations to these approaches are based on traditional vector descriptors. As pointed out in recent work [75, 76], vectorizing the original data structure, however, may destroy some inherent ordering information in the data. One example are the $n \times n$ symmetric positive semi-definite matrices. The kernel matrix in many popular kernelized machine learning algorithms [77] is of this type. Another example is the diffusion tensor (a 3×3 positive definite matrix) which is used to represent voxels in medical imaging. In computer vision, the region covariance feature, introduced in [78], is an image descriptor that captures natural correlations amongst multiple features. Hence, there has been growing interest in the development of sparse coding for positive definite descriptors. In [79], the

problem of sparse coding within the space of symmetric positive definite matrices is tackled by embedding Riemannian manifolds into kernel Hilbert spaces. [76] proposed tensor sparse coding on positive definite matrices, which keeps descriptors in their original space and uses a set of randomly selected training samples as the dictionary. It successfully extended sparse coding techniques to the space of positive definite matrices. However, little research has been done to learn a discriminative and compact dictionary over such spaces.

We present a discriminative dictionary learning method for tensor sparse coding. Rather than simply using a subset of region covariance descriptors for training images as the dictionary [76], we learn a discriminative dictionary from the training set. A structural incoherence term is introduced into the dictionary learning process to regularize the incoherence between different sub-dictionaries, which increases the discriminativeness of the learned dictionary. We further incorporate classification error into the objective function to make the learned dictionary effective for classification tasks. Instead of learning multiple classifiers for each pair of classes [39, 43, 57], a linear multi-class classifier can be easily obtained during the training process. Unlike [80], which focuses on the reconstructive capability of a dictionary, the dictionary learned by our approach has both good reconstruction and discrimination capabilities. Based on this learned high-quality dictionary, we are able to obtain discriminative tensor sparse representations. Classification can be efficiently performed on these representations using the learned multi-classifier as it only involves matrix multiplication.

4.1.1 Related Work

The region covariance descriptor was first proposed in [78] to encode an image region. The descriptor is the covariance matrix of the d-dimensional feature vectors at each pixel within a region. Given an image I, let Φ define a function that extracts a d-dimensional feature vector z_i from each pixel $i \in I$, i.e. $\Phi(I, x_i, y_i) = z_i$, where $z_i \in R^d$, and (x_i, y_i) is the location of the i^{th} pixel. Φ can be any mapping such as intensity, gradient, filter responses, etc. F is the $W \times H \times d$ dimensional features extracted from I, i.e. $F(x, y) = \Phi(I, x, y)$. For a given rectangular region $R \subset F$, $\{z_k\}_{k=1,2,...,N}$ is the set of d-dimensional features of all N points inside the region R. Then the region covariance descriptor $C_R \in R^{d \times d}$is computed by:

$$C_R = \frac{1}{N-1} \sum_{k=1}^{N} (z_k - \mu)(z_k - \mu)^T \tag{4.1}$$

where μ is the mean of all points. The region covariance descriptor fuses multiple features which might be naturally correlated to describe a region or cuboid in images or videos. The average filter during covariance computation also helps to filter out noise that corrupts individual samples. It has become a popular descriptor for human detection [79], tracking [79], object detection [80, 81], action recognition [82], and pedestrian detection [83].

The tensor sparse model introduced in [76] learns a sparse representation over positive definite matrices. In [81], the Stein kernel is introduced to embed the space of symmetric positive definite matrices into a kernel Hilbert space. These algorithms, however, take the entire training set as the dictionary. Tensor sparse coding with a large dictionary is computationally expensive when the number of training samples is large. Hence

learning compact dictionaries for tensor sparse coding is desirable. In [80], a dictionary learning method is developed based on the K-SVD algorithm [42]. However, the dictionary atoms are updated independently, and the updating aims to reduce reconstruction errors. So the learned dictionary may not perform well for classification tasks.

Compared to previous work, our approach learns a discriminative and reconstructive dictionary effectively. With respect to this dictionary, discriminative sparse representations can be obtained by solving a determinant maximization (MAXDET) problem. We simultaneously train a linear classifier along with dictionary learning, resulting in a learned dictionary good for classification.

4.1.2 Notation

$\mathbb{S}_d^+$ denotes the space of $d \times d$ symmetric positive semi-definite matrices, while $\mathbb{S}_d^{++}$ refers to the space of strictly positive definite matrices. $A \succ 0$ $(A \succeq 0)$ means A is positive (semi)definite. $A \succ B$ $(A \succeq B)$ indicate that $(A - B)$ is positive (semi)definite. Let $S = \{S_l\}_{l=1}^N$ denote the data set, $S_l \in \mathbb{S}_d^{++}$. K is the number of categories. Then the dictionary is $A = [A^1, A^2, ..., A^K]$. $A^i = [\mathbf{a}_1^i, \mathbf{a}_2^i, ..., \mathbf{a}_{K_i}^i]$ denotes the sub-dictionary for class i. K_i is the number of atoms within that sub-dictionary, and each dictionary atom $\mathbf{a}_t^i \in \mathbb{S}_d^{++}$. $X = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N] \in R^{K \times N}$ represents the sparse representation for S, with $\mathbf{x}_l$ for S_l. Then the reconstructed data $\hat{S}$ is:

$$\hat{S} = \quad X \otimes A \tag{4.2}$$

$$\mathbf{x}_l \otimes A = \quad \sum_{i=1}^{K} x_l^i \otimes A^i = \sum_{i=1}^{K} \sum_{t=1}^{K_i} x_{tl}^i \mathbf{a}_t^i \tag{4.3}$$

with x_l^i denoting the representation coefficients for S_l corresponding to sub-dictionary A^i, and x_{tl}^i is the coefficient corresponding to dictionary atom $\mathbf{a}_t^i$.

The LogDet divergence $D_{ld} : \mathbb{S}_d^+ \times \mathbb{S}_d^{++} \to \mathbb{R}_+$ is defined as:

$$D_{ld}(X, Y) = tr(XY^{-1}) - log\ det(XY^{-1}) - n \tag{4.4}$$

This measures the distance between two positive definite matrices [76, 84].

4.2 Tensor Sparse Coding and Dictionary Learning

In this section, we give a brief review of tensor sparse coding and algorithms for learning an over-complete dictionary. Given a dictionary A and a data set S, the tensor sparse coding problem in [76] is formulated as:

$$\begin{aligned} min_{\mathbf{x}\geq 0} \quad & D_{ld}(X \otimes \mathbf{A}, S) + \lambda||\mathbf{x}||_1 \\ s.t. \quad & 0 \preceq X \otimes \mathbf{A} \preceq S, \end{aligned} \tag{4.5}$$

where D_{ld} is the LogDet divergence defined in (4.4), and λ is the regularization parameter inducing sparsity on X. The problem can be reduced to a MAXDET problem [76] and solved by utilizing CVX [85].

In [76], the dictionary A was constructed by simply selecting a subset of the training set for the classification setting. In [80], a dictionary from the training data is learned via minimizing a reconstruction error. Each dictionary atom is updated based on a gradient descent or an alternating formulation method. Minimizing the reconstruction error in problem (4.5), however, may not be optimal for classification tasks. We will show that by introducing structural incoherence into the objective function of dictionary learning, the discriminability of the learned dictionary can be greatly improved. Meanwhile, by incor-

porating classification error into the dictionary learning process, we can obtain a linear multi-class classifier jointly, which will improve efficiency of classification performance and reduce computation time.

4.3 Discriminative Tensor Sparse Coding

To enhance the discriminativeness of tensor sparse codes, we want to learn a reconstructive and discriminative dictionary. Each sub-dictionary corresponds to one class. The dictionary will be more discriminative if each sub-dictionary is much more representative and specific to a particular class of images. Hence we explicitly encourage independence between dictionary atoms from different sub-dictionaries. We subsequently leverage the supervised label information of input signals into the optimization problem.

4.3.1 Structural Dictionary Learning 1 (SDL1)

The quality of the dictionary influences the discriminativeness of the tensor sparse representations. Updating each dictionary atoms separately does not result in sufficient discriminating information in the sub-dictionaries. Following [52,86], we introduce structural incoherence into sub-dictionary atoms. Incoherence will promote dictionary atoms from different classes to be independent from each other; thus it leads to sparse and discriminating representations for images.

Based on the above analysis, we add a structural incoherence regularization term into the objective function. Given a training data set $S = \{S_l\}_{l=1}^{N}$, we will learn a dictionary $A = \{A^i\}_{i=1}^{K}$, with sub-dictionary $A^i = [\mathbf{a}_1^i, \mathbf{a}_2^i, ..., \mathbf{a}_{K_i}^i]$ for class i. The problem is

formulated as:

$$
\begin{aligned}
min_{A,X} \quad & \Sigma_{l=1}^{N} D_{ld}(\mathbf{x}_l \otimes A, S_l) + \lambda||\mathbf{x}||_1 + \eta\Sigma_{i\neq j,s,t}||(\mathbf{a}_s^j)^T\mathbf{a}_t^i||_F^2. \qquad (4.6)\\
s.t. \quad & \mathbf{x}_l \geq \mathbf{0} \quad \forall l \\
& \mathbf{a}_t^i,\ \mathbf{a}_s^j \succeq 0 \quad \forall i,t,j,s \\
& 0 \preceq \mathbf{x}_l \otimes A \preceq S_l \quad \forall l
\end{aligned}
$$

The first two terms are the reconstruction error and the sparsity regularization. The last term sums up the Frobenius norms between every two dictionary atoms $\mathbf{a}_s^j, \mathbf{a}_t^i$ which belong to different sub-dictionaries A^j and A^i. λ, η are penalty parameters balancing reconstruction error, sparsity, and dictionary structural incoherence.

4.3.2 Structural Dictionary Learning 2 (SDL2)

As pointed out in [54], the learned dictionary can be more adaptive to classification tasks when minimizing the classification error in the objective function of dictionary learning. A linear multi-classifier $f(x;W) = Wx$ is used for classification. W denotes the linear classifier's parameters. Hence, the classification error can be explicitly included in the objective function during the dictionary learning. The classifier will be learned through the training process, as well. The objective function is formulated as below:

$$
\begin{aligned}
min_{A,X,W} \quad & \Sigma_{l=1}^{N} D_{ld}(\mathbf{x}_l \otimes A, S_l) + \lambda||\mathbf{x}||_1 + \eta\Sigma_{i\neq j,s,t}||(\mathbf{a}_s^j)^T\mathbf{a}_t^i||_F^2 + \gamma||H - W\mathbf{x}||_2^2 \qquad (4.7)\\
s.t. \quad & \mathbf{x}_l \geq \mathbf{0} \quad \forall l \\
& \mathbf{a}_t^i,\ \mathbf{a}_s^j \succeq 0 \quad \forall i,t,j,s \\
& 0 \preceq \mathbf{x}_l \otimes A \preceq S_l \quad \forall l
\end{aligned}
$$

where the term $||H - WX||_2^2$ represents the classification error. $H = [h_1, h_2, ..., h_N] \in R^{K \times N}$ denotes the label matrix. The column vector $h_i = [0, 0, ...1...0, 0]^T \in R^K$ is a label vector for sample i. The position of 1 indicates its class index. γ controls the contribution of the classification error regularization term in the optimization process.

4.3.3 Optimization

In this section, we only describe the optimization procedures for SDL2 here. To solve SDL1, we utilize a similar procedure excluding the calculation of classifier W. The classifier will be calculated directly through Equation (4.16) using the final result X for SDL1.

The dictionary learning problem is convex in any one of the elements in the triple (A, X, W) when the others are fixed. Hence, the optimization can be divided into three subproblems: (1) updating dictionary atoms with fixed X and W; (2) solving the max determinant problem with fixed A and W; (3) computing a linear classifier with fixed X and A. If we set $\gamma = 0$ in subproblem (2), this is exactly the optimization procedure for problem (4.6). The complete optimization is summarized in Algorithm 5.

4.3.3.1 Dictionary Update with fixed W and X

Following [80], we use a steepest descent approach to update each dictionary atom $\mathbf{a}_t^i$. With fixed W and X, the objective function (4.7) can be rewritten as a function of $\mathbf{a}_t^i$

as below:

$$f(\mathbf{a}_t^i) = \Sigma_l D_{ld}(\mathbf{x}_l \otimes A, S_l) + \lambda||\mathbf{x}||_1 + \eta\Sigma_{j\neq i}\Sigma_s||(\mathbf{a}_s^j)^T\mathbf{a}_t^i||_F^2 + \gamma||H - WX||_2^2 \quad (4.8)$$

$$= \Sigma_l tr(x_{tl}^i \mathbf{a}_t^i S_l^{-1}) - log\ det\hat{S}_j + \eta\Sigma_{j\neq i}\Sigma_s tr((\mathbf{a}_t^i)^T\mathbf{a}_s^j(\mathbf{a}_s^j)^T)\mathbf{a}_t^i + C, \quad (4.9)$$

where C includes all those terms independent of $\mathbf{a}_t^i$. When updating one dictionary atom, other atoms remain fixed. The gradient descent direction $-\nabla f(\mathbf{a}_t^i)$ is:

$$-\nabla f(\mathbf{a}_t^i) = \Sigma_l x_{tl}^i({\hat{S}_l}^{-1} - S_l^{-1}) - 2\eta\Sigma_{j\neq i}\Sigma_s(\mathbf{a}_s^j)^T\mathbf{a}_s^j\mathbf{a}_t^i \quad (4.10)$$

Since $A_i \in \mathbb{S}_d^{++}$, we need to ensure that $d\mathbf{a}_t^i \succeq 0$. Meanwhile, from $\hat{S}_j \preceq S_j$, we know that ${\hat{S}_j}^{-1} \succeq S_j^{-1}$, yielding the first term in Equation (4.10) positive semidefinite. Thus the gradient direction $d\mathbf{a}_t^i$ is given by:

$$d\mathbf{a}_t^i = \begin{cases} \Sigma_l x_{tl}^i({\hat{S}_l}^{-1} - S_l^{-1}) - 2\eta\Sigma_{j\neq i}\Sigma_s(\mathbf{a}_s^j)^T\mathbf{a}_s^j\mathbf{a}_t^i\ , & \nabla f(\mathbf{a}_t^i) \leq 0 \\ \Sigma_l x_{tl}^i({\hat{S}_l}^{-1} - S_l^{-1})\ , & otherwise \end{cases} \quad (4.11)$$

Combining all dictionary atoms together, the dictionary is updated as below. The updating step size $\alpha \geq 0$ is determined by a line search technique.

$$dA = [dA^1, dA^2, ..., dA^K] = [d\mathbf{a}_1^i, d\mathbf{a}_2^i, ..., d\mathbf{a}_{K_i}^i] \quad (4.12)$$

$$A \leftarrow A + \alpha\, dA \quad (4.13)$$

4.3.3.2 Solving X with fixed A and W

With fixed A and W, the subproblem to solve X can be formulated as:

$$min_X \quad \Sigma_{l=1}^N D_{ld}(\mathbf{x}_l \otimes A, S_l) + \lambda||\mathbf{x}||_1 + \gamma||H - WX||_2^2 \quad (4.14)$$

$$s.t. \quad \mathbf{x}_l \geq \mathbf{0}\ \forall l, \quad \mathbf{a}_t^i \succeq 0\ \forall i,t, \quad 0 \preceq \mathbf{x}_l \otimes A \preceq S_l\ \forall l$$

This is a problem of sparse decomposition over positive definite matrices. As shown in [76], this problem is convex and well-behaved. It falls into the general class of opti-

mization problems known as MAXDET problems. CVX [85] is used to solve this problem.

4.3.3.3 Calculating W with fixed A and X

We use the multivariate ridge regression model [50,69] to obtain the linear classifier W:

$$\hat{W} = \arg\min_{W} ||H - WX||_2^2 + \lambda_w ||W||_2^2 \tag{4.15}$$

where H is the class label matrix of X. Fixing X and A, the multi-class classifier can be easily derived as:

$$W = HX^T(XX^T + \lambda_w I)^{-1} \tag{4.16}$$

Algorithm 5 Structural Dictionary Learning

Input: Data S, and Parameters α, λ, η, γ

Output: A, W

Initialize: Initial Dictionary A, Classifier W, $\epsilon_A = 10^{-3}$

while not converged, $i \leq maxIterA$ **do**

fix A and W, solve X for MAXDET problem (4.14)

fix A and X, calculate W according to (4.16)

fix W and X, calculate dA according to (4.11)

update A with (4.12) and (4.13)

check the convergence condition:

$||S - X \otimes A||_\infty < \epsilon_A$

4.3.3.4 Initialization

To initialize the dictionary A, we randomly sample A from the training data. The initialization of subdictionary A_i is a subset of training data belongs to class i. To initialize W, we first solve problem (4.14) with $\gamma = 0$ using the initialized dictionary. Then W is calculated according to Equation (4.16).

4.3.4 Classification

After obtaining the dictionary, a tensor sparse representation X_{test} for test data S_{test} is calculated by solving Equation (4.14) with $\gamma = 0$. The representation $\mathbf{x}_l$ for test sample l is the l-th column in X_{test}. Using the classifier W, the label for test sample l is given by $k = arg\max_k(\mathbf{s} = W\mathbf{y}_l)$ which corresponds to the index of the largest element in the class label vector $\mathbf{s}$.

4.4 Experiments

We evaluate our approach on three datasets: Brodatz texture dataset [87], USPS digital dataset [88], and AR face dataset [72]. Our approach is compared with several state-of-the-art algorithms including tensor sparse coding (TSC) [76], tensor sparse coding with dictionary learing (TSCwD) [80], logE-SR [82] and Riemannian sparse representation (RSR) [81].

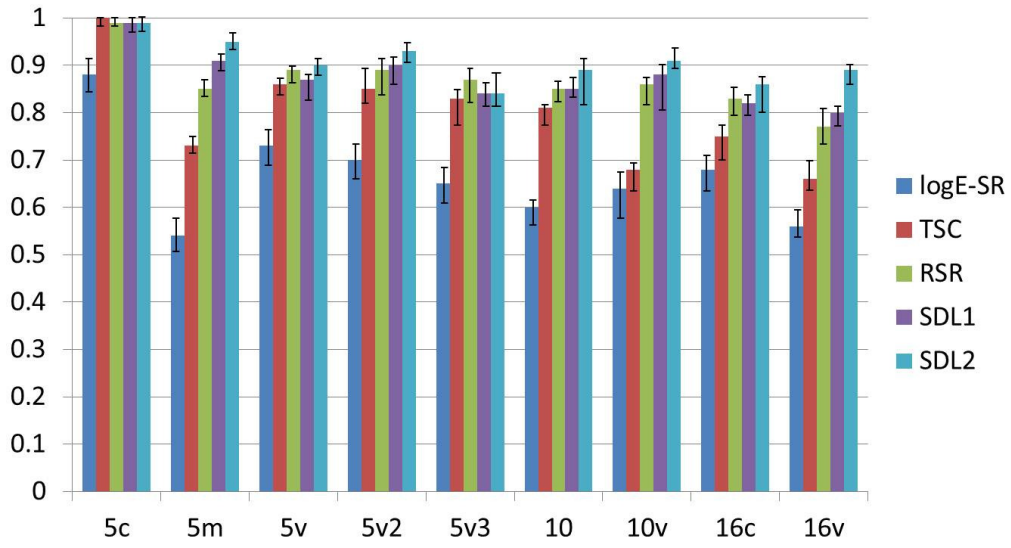


Figure 4.1: Texture classification results on the Brodatz dataset. Each group (five bars) indicates the recognition accuracy for one test scenario. Each bar in a group corresponds to one method.

4.4.1 Texture Dataset

We follow the protocol in [81] and create mosaics under nine test scenarios with various number of classes, including 5-textures ('5c', '5m', '5v', '5v2', '5v3'), 10-textures ('10', '10v'), and 16-textures ('16c', '16v'). Spatial derivatives have been shown to be useful for texture characterization in [76,78]. The feature vector $F(x, y)$ for any pixel with gray scale intensity $I(x, y)$ is $[I(x, y),\ |\frac{\partial I}{\partial x}|,\ |\frac{\partial I}{\partial y}|,\ |\frac{\partial^2 I}{\partial x^2}|,\ |\frac{\partial^2 I}{\partial y^2}|]$. Each image is 256×256, and 32×32 blocks are cut out, yielding 64 data samples per image; a 5×5 region covariance descriptor is computed for each sample. For each scenario, we randomly select 5 data samples as training and the rest for testing. The random selection is repeated 10 times.

Figure 4.1 illustrates the classification results under different testing scenarios. We

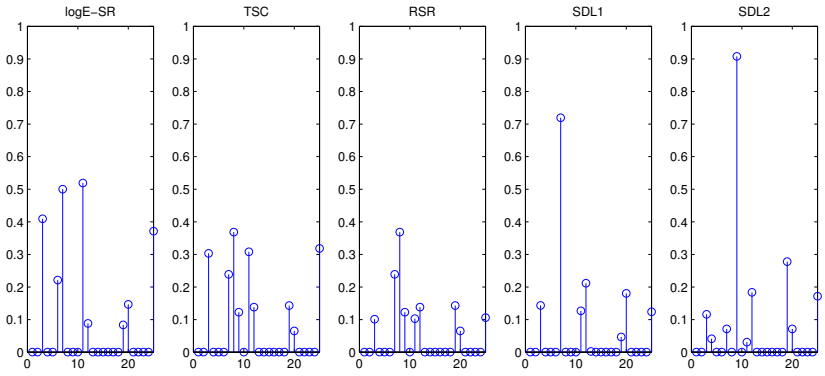


Figure 4.2: An example of tensor sparse codes using different approaches. X axis indicates the dimension of sparse codes, Y axis indicates the average tensor sparse codes for testing samples (first 10 blocks) from the 2nd class in '5v2'.

compare SDL1 and SDL2 against logE-SR [82], TSC [76,80], and RSR [81]. The average accuracy of SDL2 achieves the highest score on all test scenarios except for '5v3' and '5c'. We use $\alpha = 0.0001, \lambda = 0.001, \eta = 0.0001$ in our experiments. However, our maximum classification results over 10 runs are comparable to the best scores. Figure 4.2 shows an example of tensor sparse codes on '5v2'. The indices 6$\sim$10 on the X-axis corresponds to the sub-dictionary for the 2nd class. The coefficients highly peak within the 2nd class in our method. We can see that SDL2 provides the most discriminative sparse codes among all methods.

4.4.2 Digit Dataset

The USPS Dataset is a handwritten digit database containing 9298 16×16 handwritten digit images. We follow the protocol in [68], using the images of digits 1,2,3,4 and randomly selecting 200 images for each category. The percentage of training samples ranges from 10% to 60%. For tensor sparse coding methods, a 9×9 covariance descriptor

is used to describe a digit image, using the feature below:

$$F(x,y) = [I(x,y),\ |G_{0,0}(x,y)|,\ ...,\ |G_{0,3}(x,y)|,\ |G_{1,0}(x,y)|,\ ...,\ |G_{1,3}(x,y)|] \quad (4.17)$$

where $I(x,y)$ is the intensity value at position (x,y) and $G_{u,v}(x,y)$ is the response of a 2D Gabor wavelet [89] centered at (x,y) with orientation u and scale v:

$$G_{u,v} = \frac{k_v^2}{4\pi^2} \sum_{t,s} e^{-\frac{k_v^2}{8\pi^2}((x-s)^2+(y-t)^2)} (e^{ik_v((x-t)cos(\theta_u)+(y-s)sin(\theta_u))} - e^{-2\pi^2}) \quad (4.18)$$

with $k_v = \frac{1}{\sqrt{2^{v-1}}}$ and $\theta_u = \frac{\pi u}{4}$.

Table 4.1 summarizes classification performances using different approaches. The results of kNN0, kNN1, NNLRS-graph [68] are copied from [68]. It can be seen that our discriminative tensor sparse coding method is comparable to other methods and outperforms the previously proposed dictionary learning method for tensor sparse coding [80].

In Figure 4.3(b), we illustrate how classification errors decrease with the number of iterations among dictionary learning methods. As expected, the error rate of SDL2 decreases the most rapidly compared to the parallel-sum method introduced in [80] and SDL1.

4.4.3 Face Dataset

The AR face database includes over 4,000 color face images of 126 individuals, 26 images for each person in two sessions. The images are cropped to 27×20 and converted into gray scale. The images from 10 subjects are used in our experiment. The images are convolved with Gabor filters using Equation (4.18) with 8 orientations $\theta_u = \frac{\pi u}{8}$, $u = 0, 1, ..., 7$, and up to 5 scales $v = 0, 1, 2, 3, 4$. We use the same features as [81] for face recognition. Each face image is described with a 43×43 covariance descriptor

using the following features:

$$F(x,y) = [I(x,y),\ x,\ y,\ |G_{0,0}(x,y)|,\ ...,\ |G_{0,7}(x,y)|,\ |G_{1,0}(x,y)|,\ ...,\ |G_{4,7}(x,y)|] \quad (4.19)$$

where $I(x,y)$ is the intensity at (x,y) and $G_{u,v}$ is the response of a 2D Gabor wavelet.

We compare our methods with other covariance descriptor based methods including TSC [76], TSC with dictionary update [80], and RSR [81]. The learned dictionary has 7 dictionary atoms per person. For each person, we randomly select 15, 18, 21 images for training and the remainder for testing. Table 4.2 summarizes the experimental results. SDL2 obtains the best performance in this experiment.

4.5 Conclusion

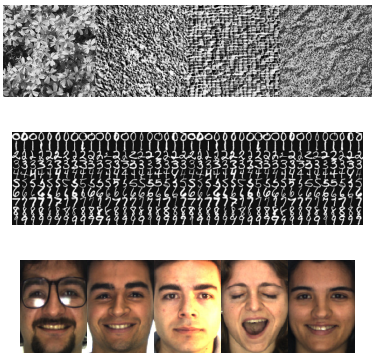
We introduced a discriminative dictionary learning approach for tensor sparse coding. The introduction of structural incoherence between dictionary atoms from different sub-dictionaries encourages disparity among sub-dictionaries, thus enhancing discriminating ability of the sparse representation. We further incorporate label information into the optimization problem so that the learned dictionary is more useful for classification. The SDL1, SDL2 problems can be formulated as MAXDET problems and the dictionary atoms can be updated through gradient descent. Experimental results demonstrate that our approach is robust and effective.

Table 4.1: Classification error rates (%) using different approaches with different sampling percentages

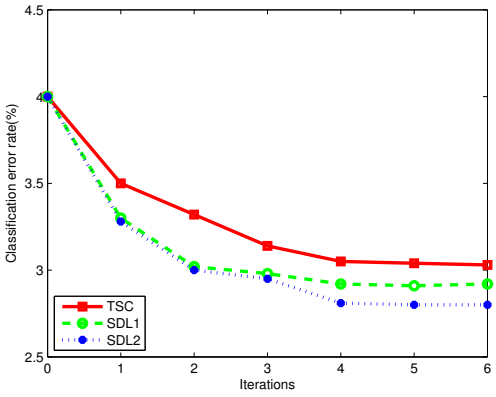
Database	kNN0	kNN1	NNLRS [68]	TSCwD [80]	SDL1	SDL2
USPS (10%)	3.13	3.21	2.80	3.03	2.92	2.80
USPS (20%)	2.22	2.10	1.62	1.98	2.02	1.65
USPS (30%)	1.55	1.53	1.13	1.20	1.56	1.02
USPS (40%)	1.20	1.18	0.88	1.01	0.94	0.82
USPS (50%)	0.82	0.86	0.59	2.80	0.58	0.49

Table 4.2: Recognition rates on the AR face database

number of train samp.	TSC [76]	TSCwD [80]	RSR [81]	SDL1	SDL2
15 per person	70.2	78.6	81.4	80.0	82.3
18 per person	73.5	79.9	84.1	82.0	85.2
21 per person	75.8	80.8	85.7	83.2	86.1



(a) Sample images



(b) Classification error

Figure 4.3: (a) The first row shows sample images from the Brodatz texture dataset; the second row shows sample images from the USPS dataset; the third row shows sample images from the AR dataset. (b) Classification error for different dictionary learning algorithms. For this experiment, we use 20 training images per class.

Chapter 5: Unsupervised Abnormal Crowd Activity Detection Using Semi-parametric Scan Statistic

5.1 Overview

Activity recognition is another challenging task in computer vision system. In the chapter, we address the problem of abnormal activity detection. Identifying abnormal activities in densely crowded scenes has been attracting increasing attention in the computer vision community. This problem plays an important role in many applications such as crowd surveillance, public place monitoring, security control, etc. The main paradigm in this field is to assume the availability of a set of normal examples before detection, which define what normal activity look like. Abnormality of a new observation is then measured either by its similarity with the given examples or by its compatibility with the model derived from the examples.

The requirement of the normal activity begin defined beforehand may complicate the deployment of the approaches in real applications. For example, an important characteristics of normal and abnormal activities in practice is their relativity. Abnormal activities in one situation may become normal in others. Fig 5.1 shows a scene in a subway station. The gate is used as an exit gate and people entering it is regarded as abnormal.

Figure 5.1: Examples of abnormal activities in crowded scenes. (a) Temporal anomalous activity of crowd panic. (b) Spatial-temporal anomalous activity where a man enters through the exit gate in a subway station.

However, at other times the same gate may be changed to an entrance gate and people entering it becomes normal while people exiting through it becomes abnormal. To apply detection approaches based on the above paradigm, one must know in advance whether the gate is served as entrance or exit gate at a specific time, so that the correct normal activity model can be used.

To reduce the requirement for external examples of normal activity, some recent work assumes that a certain duration in the beginning of a video contains only normal behaviours, which are used to train the normal activity model [90–92]. However, these methods cannot be applied when the normal behaviours keep changing over time so that the behaviors in the beginning of a video cannot be used to infer thsoe in the remaining of the video. They are not applicable when the abnormal behaviours appear right after the video begins.

To address the above limitations of previous approaches, we propose a fully un-

supervised approach, which requires neither normal nor abnormal examples being provided beforehand, for anomaly detection in crowded scenes. In this scenario, normal activities are the behaviours performed by majority of people and abnormal activities are behaviours that occur rarely and are different from most others in the scene. Based on this characteristic of the anomaly in crowded scenes, we propose to use a scan statistic method to solve the problem. The main idea is to scan the video using a large number of windows with variable sizes. For each window, two hypotheses about whether or not the characteristics of the observations inside the window is different from those outside it are compared. A likelihood ratio test statistic, which is based on a semi-parametric density ratio model, is computed for the comparison and used as a measure of the window's abnormality.

Besides the waiver of the requirement for normal activity examples, the novelty and advantages of the scan statistic method also lie in the following aspects. First, as a quite general detection framework, it can be used to detect different kinds of abnormal events, including temporal abnormalities (also called global abnormal event), where the whole crowd are involved in the abnormal event in which their behaviours are different from those at most other times in the video (5.1(a)); and spatial-temporal abnormalities, where the local behaviours are different from most others in the entire video (5.1(b)). Second, in previous scanning window based approaches [90, 92], the size of the local windows are usually fixed and the abnormalities are measured independent of another. In this scan statistic method, variable sized windows are used so that an abnormal event can be involved in a single window instead of being divided up, which allows its statistic characteristics being measured more accurately. Third, a semi-parametric density ratio

method [93] is used to model the observations inside and outside a scanning window. This reduces the requirement for assuming a specific parametric probability model and makes it directly applicable to different types of observations. Fourth, to reduce the computational complexity of exhaustive search, we present a fast scanning algorithm for the semi-parametric scan statistic method. Its validity is theoretically proved and also verified through experiment.

Scan statistics are a powerful method for cluster detection. They have applications in many fields, such as epidemiology, criminology, genetics, mining, astronomy, etc. However, their use for solving computer vision problems has not been exploited before. In this paper, we illustrate its utility for abnormal activity detection in crowded scenes. Experiments on both benchmark datasets and videos "in the wild" validate the effectiveness of the method.

5.2 Related Work

Various approaches have been proposed for anomaly detection. They can be broadly categorized based on whether or not examples of normal and abnormal activities are needed before detection. The first type of work treats the task as a binary classification problem. To train the classifier, not only normal but also abnormal activities are needed. In [94, 95], both abnormal and normal activities were used to train the support vector machine (SVM) for abnormal activity recognition.

Considering the rich patterns of irregular behaviours, the second type of work finds abnormalities without knowing what kind of abnormalities will happen beforehand. Al-

though abnormal training samples are not needed, training data are still needed to define what normal activities look like. Note that since abnormal examples are not used during training, some approaches [92,96,97] have claimed to be unsupervised. According to how normal examples are used, the methods can be further divided into two sub-categories. The first one contains data-driven methods. They directly compare new observations with a set of normal examples. The observations whose similarity scores are low [97] or which cannot be composed well by the know examples [98] are regarded as abnormal. In the second sub-category, normal examples are used to build explicit models for normal activities. Anomaly is declared when the new observation cannot be explained well by the model. Kim and Grauman used Mixture of Probabilistic PCA to model normal local activity patterns. In [99], Latent Dirichlet Allocation(LDA) was used to discover latent topics in normal activities. With the recent popularity of sparse coding, normal basis sets are also learned from normal activities over which sparse reconstruction costs are computed for new observations [90, 92].

Our work belongs to the third type where neither normal nor abnormal examples are required before detection. Several such approaches have been presented, mainly for non-crowded scenes [98, 100]. the main idea of both [100] and [98] is to use the input video itself to build the reference database and compare each event with all other events observed. However, for crowded scenes, performing such pairwise comparison is time-consuming. considering the homogeneous characteristic of the activities of the crow, our method collectively model the whole crowd's behaviours, which is more appropriate and efficient for crowded scenes.

Among recent work that focuses on crowded scenes, much effort has been de-

voted to design descriptive features to characterize crowded scenes, such as social force model [99], chaotic invariants of particle trajectories [101] and mixture of dynamic textures [102]. Instead of designing new video representation, we propose a new detection framework which can be combined with different representations. We show that competitive performance can be achieve even only using traditional optical flow as the descriptors.

5.3 Anomaly Detection with Scan Statistic

5.3.1 The Scan Statistic Method

To discover spatial-temporal regions in a video in which the activities are abnormal, a sliding window with variable size is applied to the video to examine each possible region. This is similar to the sliding window based object detection framework, although the window in that case moves within an image and only the spatial size of the window is varied.

For abnormal event detection, the window is three-dimensional and therefore more flexible. For temporal anomaly, where the whole crowd are involved in the abnormal event in which their behaviors are distince from those at most other times, we fix the spatial size of the window to cover the whole image but allow its temporal length to be variable. The window moves along the time axis to examining sets of continuous frames. For spatial anomaly, where the behaviours of a small group of people are different from those of all the others observed at the same time. We let the temporal length of the window cover a whole video clip. The spatial size of the window will be variable so the window moves around in the spatial space to detect local abnormalities. This scanning method is

applicable when the activities of the whole crowd are highly dynamic and change over-time. We first divide the video into small clips and detect spatial anomaly for each clip independently. For spatial-temporal anomaly, where the crowd's normal activities are relatively consistent and the abnormalities are local behaviours that are different from most others in the entire video, both spatial and temporal size of the window can change when it moves around in the three-dimensional spatial-temporal space.

For each window S, we measure whether the observation in it is anomalous. We take the whole video as the study region to find the abnormalities. Two hypotheses about the observations in the study region are defined. The null hypothesis H_0 assumes no anomaly exists in all the observations, $i.e.$, the underlying characteristics of observations throughout the whole video is the same. The alternative hypothesis $H_1(S)$ assumes that the characteristic of observations inside window S are different from that outside S. The likelihood ratio test statistic $\lambda(S)$ defined as below is used to decide an anomaly.

$$\lambda(S) = \frac{Pr(Data|H_1(S))}{Pr(Data|H_0)} \tag{5.1}$$

where $Pr(Data|H_1(S))$ is the likelihood function computed based on the chosen probility model and the observed data under the hypothesis H_i.

The likelihood ratio test is a measure of the strength of H_1. The larger this number is, the more likely H_1 is true and the observation inside S is abnormal. Therefore, we can use $\lambda(S)$ as a measure of the abnormality of a window.

5.3.2 Semi-parametric Density Ratio Model

To compute the likelihood ratio test statistic in 5.1 for each scanning window, we use a semi-parametric density ratio model [93] to describe the probabilistic model of the observation data. A certain window S separates the whole study region into two parts. This results in one sample set x_1 associated with region within S and one sample set x_2 associated with the region outside S.

$$\mathbf{x}_1 = [x_{11}, x_{12}, ..., x_{1n_1}]^T \tilde{g}_1(x) \tag{5.2}$$

$$\mathbf{x}_2 = [x_{21}, x_{22}, ..., x_{2n_2}]^T \tilde{g}_2(x)$$

where x_1 contains the samples inside window S with sample size n_1, and x_2 contains the samples outside window S with sample size n_2. $g_i(x)$ is the probability density function of $x_{i,j}$, $i = 1, 2$; $j = 1, ..., n_i$. Taking the sample outside the scanning window as the reference density, the density ratio model assumes the ratio between the density inside S and the reference has an exponential form:

$$\frac{g_1(x)}{g_2(x)} = exp(\alpha + \boldsymbol{\beta}^T \mathbf{h}(x)) \tag{5.3}$$

Here $\mathbf{h}(x)$ is predefined function of x which can take forms such as x, x^2, $log(x)$, or $[x, x^2]^T$. α is a scalar, $\boldsymbol{\beta}$ can be a scalar or vector based on $\mathbf{h}$.

Obviously $\boldsymbol{\beta} = \mathbf{0}$ implies $\alpha = 0$ and thus we have $g_1(x) = g_2(x)$, which means the samples inside and outside the window S come from the same distribution. Therefore, the null hypothesis of the no anomaly can be represented as H_0: $\boldsymbol{\beta} = 0$.

Note we do not have to know the exact form of the probability density function $g_1(x)$ and $g_2(x)$. The density ratio model (Eqn 5.3) applies to different distribution by

varying **h**. For example, for Bernoulli and Poisson distributions, we have $h(x) = x$. Such model greatly reduce the need to know prior distribution information and applies to more generalized conditions.

5.3.2.1 Parameter Estimation

We estimate the parameters α and $\boldsymbol{\beta}$ using the data in the study region. Following the profiling procedure [103], we have the log-likelihood up to a constant as a function of α and $\boldsymbol{\beta}$.

$$l(\alpha, \boldsymbol{\beta}) = -\sum_{i=1}^{n} log[1 + \rho exp(\alpha + \boldsymbol{\beta}^T \mathbf{h}(s_i))] + \sum_{j=1}^{n_1} [\alpha + \boldsymbol{\beta}^T \mathbf{h}(x_{1j})] \tag{5.4}$$

where $\rho = \frac{n_1}{n_2}$. α and $\boldsymbol{\beta}$ are estimated by maximizing this log-likelihood (Eqn5.4). The maximum estimate are denoted as $\hat{\alpha}$ and $\hat{\boldsymbol{\beta}}$.

5.3.2.2 Computing the Likelihood Ratio Test Statistic

To compute the abnormality of each window S, we use the likelihood ratio test (LR-test) $\lambda(S)$. It can be formulated as below:

$$\begin{aligned} \lambda(S) &= -2[l(0, \mathbf{0}) - l(\hat{\alpha}, \hat{\boldsymbol{\beta}})] \\ &= -2\sum_{i=1}^{n} log[1 + \rho exp(\hat{\alpha} + \hat{\boldsymbol{\beta}}^T \mathbf{h}(s_i))] + 2\sum_{j=1}^{n_1} [\hat{\alpha} + \hat{\boldsymbol{\beta}}^T \mathbf{h}(x_{1j})] + 2nlog(1 + \rho) \end{aligned} \tag{5.5}$$

We use a tunable threshold for $\lambda(S)$ to determine when an abnormal event is detected. Similar to the object detection, the performance is evaluated at multiple thresholds which trade off accuracy and false alarm rate.

The semi-parametric density ratio method has two major advantages. First, the parameters α, β and the distributions are estimated from the combined data, not just from

samples either inside or outside the scanning window. This avoids the possible difficulty of estimating the parameters only from samples inside the window when the window size is small. Second, the method does not require assumptions about the specific parametric probability model of the data. Therefore, the same method can be directly applied to problems with quite different data distributions.

5.4 A Fast Scanning Algorithm

5.4.1 Fast Scan with Windows of Fixed Size

One major challenge of the scan statistic method is the large number of regions that need to be scanned. It is generally computationally infeasible to search all of them. It is common to use domain knowledge to restrict the search space.

For abnormal activity detection in video, this problem is even more severe. Due to the dynamic nature of human motion, the abnormal activity usually occupies irregular regions in the three-dimensional space. This requires more flexible shapes for the scanning window. Moreover, there are usually multiple anomalies in the video. This makes methods that only search for the most anomalous region inapplicable. In this section, we propose an efficient algorithm for the density ratio model based scan statistic method. We observed that for windows where the numbers of samples in them are equal, the following theorem holds.

Theorem 1 For windows with the same inside window sample size n_1, those that only contain anomalous samples have larger $\lambda(S)$ than the others.

Intuitively, consider two windows where the first one only contains abnormal data

and the the second only contains normal data. When comparing the data distributions inside and outside the windows, for both case the outside is a mixture of a large number of normal samples together with a small number of abnormal samples. It is obvious that the discrepancy between inside and outside window distributions are larger for the first window than for the second one.

Since any abnormal region can be regarded as a combination of a set of equal sized subregions, instead of searching for the region with windows of variable size, we can first scan the whole study region with a single-sized window. Then we rank the windows in descending order of their likelihood ratio test $\lambda(S)$. According to **Theorem 1**, windows that are located in abnormal rgions will always rank high in the list. We take the top ranked windows, and merge those that overlap or nearby with each other. This generates several bigger regions with different shape and size.

In order for **Theorem 1** to hold, we have assumed that the probability distributions inside and outside abnormal regions are homogeneous respectively. However, real videos are complex. There are fluctuations both spatially and temporally in them. Therefore, the likelihood ratio test scores for some sub-areas inside the abnormal region be smaller than some areas located in the normal region. In this case, the regions detected by the fast scanning algorithm may not be exactly the same as the exhaustive search scheme. We found through experiment that the difference is not significant. Therefore considering efficiency, we prefer this fast scanning algorithm when analysing large scale video data.

Figure 5.2: Normal activities of the three crowded scenes of UMN dataset.

5.5 Experimental Results

In this section, we show the performance of the proposed abnormal activity detection algorithm on tow benchmark datasets and several videos ”in the wild” that are downloaded from the web. The three datasets show examples of the three kinds of abnormal events, i.e. temporal, spatial and spatial-temporal anomalies, respectively.

5.5.1 UMN Dataset

We use the UMN dataset to evaluate the performance for temporal anomaly (global abnormal event) detection. The dataset includes 11 video sequences of three different scenes of crowded escape events. Each video begins with normal behaviour where people work around, followed by a sudden abnormal panic. Fig 5.2 shows the normal behaviours of the three scenes.

Since the whole crowd are involved in the abnormal activities, we can fix the spatial size of the scanning window to cover the whole image and only need to detect in which frames the activites are anomalous. The number of candidate windows is $O(t^2)$, where t is the number of frames in a video. In this situation, scanning all windows is feasible.

We therefore are able to compare the results of exhaustive search and the fast scanning algorithm in Section 5.4. For exhaustive search, the window with the highest $\lambda(S)$ is chosen. For fast scanning, the window length is set to 20 frames in the first round of search. After merging the top windows and recomputing $\lambda(S)$ for the big windows, the window with the highest $\lambda(S)$ is reported. The step size for moving the windows is set to 5 for both methods.

To quantitatively compare the performance of our algorithm with other state-of-the-art methods, we draw ROC curve for frame-level measurement. Table 5.1 compares the corresponding average AUC with other methods. Our scan statistic method achieved very competitive performance. Note that all other methods require examples of normal activities being provided, which are used either to train their models or as references for comparison. Our method does not have this requirement. No training or initialization are involved. The result demonstrates the effectiveness of the scan statistic method for global abnormal events detection in videos.

5.5.2 Subway Surveillance Data

We use the subway surveillance videos in [104] to evaluate the performance for spatial-temporal anomaly detection. There are two videos in the dataset. One video monitors the entrance gate, which is 1 hour 36 minutes long with 144,249 frames in total. The other watches the exit gate and is 43 minutes long with 64,900 frames. We follow the same definition of abnormal activities used in [91]. Specifically, the following abnormal activities are defined: (a) Wrong direction (WD): people exit through entrance gate or

Method	Area under ROC
Chaotic invariant [101]	0.99
Social force [99]	0.96
Optical flow [99]	0.84
Nearest neighbor [90]	0.93
Sparse Reconstruction [90](Scene1/Scene2/Scene3)	0.995/0.975/0.964
Our method(Scene1/Scene2/Scene3)	0.991/0.951/0.99

Table 5.1: The comparison of our semi-parametric scan statistic method with other methods on the UMN dataset.

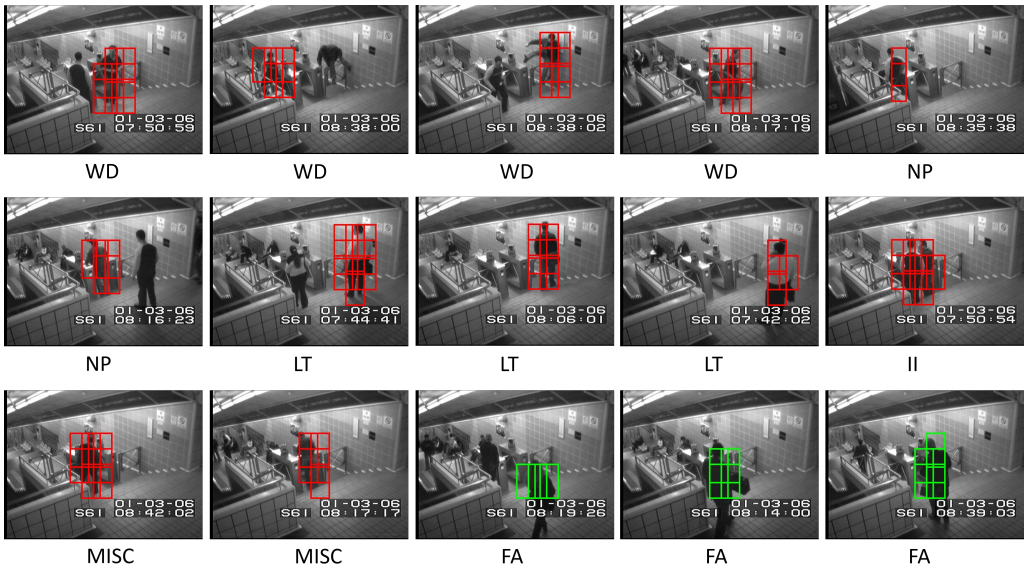


Figure 5.3: Example of abnormal activities detected in subway entrance video, including wrong direction (WD), loitering (LT), irregular interactions (II), misc. and false alarm (FA). We show the merged windows which consist of multiple overlapping sub-windows of fixed size. False alarms are marked with green windows.

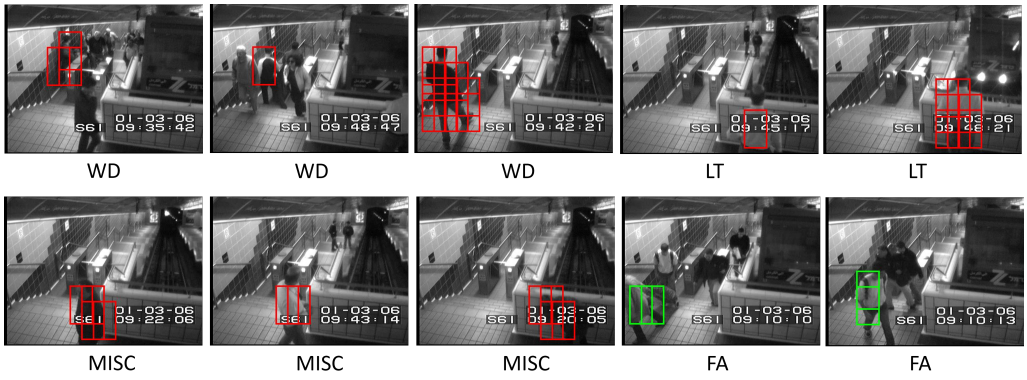


Figure 5.4: Example of abnormal activities detected in subway exit video, including wrong direction (WD), loitering (LT), misc. and false alarm (FA). We show the merged windows which consist of multiple overlapping sub-windows of fixed size. False alarms are marked with green windows.

enter through the exit gate; (b) No payment (NP): people enter the entrance gate without payment; (c) Loitering (LT): people loiter at the station; (d) Irregular interactions between persons (II); (e) Misc: e.g. a person suddenly stops walking, or runs fast. The entrance gate video only includes the wrong direction, loitering and misc. events.

In this experiment, we use histogram of optical flow (HoF) to represent the local optical flow patch. The HoF features [105] are then quantized to flow words. The spatial size of the window is fixed to 40×90 pixels and the length is set to 40 frames during the first round of scan. We then compare the detected merged regions with the ground truth. The quantitative comparison of the result with other methods is shown in Table 5.5.2. We can see that our algorithm achieved similar performance as other state-of-the-art methods. Unlike our method, both of the other two methods use video clips containing normal activites in the first few minutes of the video to train their models. Our method start

Method	WD	NP	LT	II	Misc.	Total	FA
Ground truth	26/9	13/-	14/3	4/-	9/7	66/19	0/0
ST-MRF [90]	24/9	8/-	13/3	4/-	8/7	57/19	6/3
Sparse Coding [92]	25/9	9/-	14/3	4/-	8/7	60/19	5/2
Our method	26/9	6/-	14/3	4/-	8/7	58/19	6/2

Table 5.2: Comparison of abnormal activity detection result on subway surveillance data. Abnormal activities include wrong direction (WD), no payment (NP), loitering (LT), irregular interaction (II) and misc. FA stands for false alarm. The number before the slash(/) denotes the entrance gate result, and the number after it is for the exit gate result.

detection without training so that not knowing whether the gate is for entrance or exit beforehand. This is mined directly from the testing videos.

Our method missed half of the no payment event. This may due to the fact that the gate is located far from camera and people turn their back towards the camera during paying. Additionally, some no payment actions are too subtle to be recognized. We show some examples of the detected abnormal activities in Fig 5.3 and Fig 5.4. The detected regions are composed of several overlapping sub-windows of fixed size. The composed windows show good coverage of the abnormal events.

Appendix A: Previous Proof

Proof of generating discriminative object proposals via submodular ranking.

A.1 Proof of Submodularity of the Weighted Coverage Term H(A)

Recall our definition of the weighted coverage term $H(A)$.

$$H(A) = \sum_{i \in V} \max_{j \in A} w_{ij} - \sum_{j \in A} \phi_j \tag{A.1}$$

where V is the set of all segments. Each vertex $i \in V$ is an element, and the weight w_{ij} measures the appearance similarity between vertices i and j. ϕ_j is the cost of adding element j to A.

Proof. From equation A.1, we have the marginal gain for element a given a selected set A.

$$\begin{aligned}
& H(A \cup \{a\}) - H(A) \\
= & \sum_{i \in V} \max_{j \in A \cup \{a\}} w_{ij} - \sum_{j \in A \cup \{a\}} \phi_j - \sum_{i \in V} \max_{j \in A} w_{ij} + \sum_{j \in A} \phi_j \\
= & \sum_{i \in V} \max(\max_{j \in A} w_{ij}, w_{ia}) - \sum_{i \in V} \max_{j \in A} w_{ij} - \phi_a \\
= & \sum_{i \in V} \max(w_{ia} - \max_{j \in A} w_{ij}, 0) - \phi_a
\end{aligned}$$

For $\forall\ A_1 \subseteq A_2 \subseteq V$ and any element $a \in V \setminus A_2$, we can compute:

$$(H(A_1 \cup \{a\}) - H(A_1)) - (H(A_2 \cup \{a\}) - H(A_2))$$

$$= \sum_{i \in V} \max(w_{ia} - \max_{j \in A_1} w_{ij}, 0) - \phi_a - \sum_{i \in V} \max(w_{ia} - \max_{j \in A_2} w_{ij}, 0) + \phi_a$$

$$= \sum_{i \in V} \max(w_{ia} - \max_{j \in A_1} w_{ij}, 0) - \sum_{i \in V} \max(w_{ia} - \max_{j \in A_2} w_{ij}, 0)$$

As $A_1 \subseteq A_2$, then we have $\max_{j \in A_1} w_{ij} \leq \max_{j \in A_2} w_{ij}$, thus $\max(w_{ia} - \max_{j \in A_1} w_{ij}, 0) \geq \max(w_{ia} - \max_{j \in A_2} w_{ij}, 0)$ for $\forall i \in V$. Therefore, $(H(A_1 \cup \{a\}) - H(A_1)) - (H(A_2 \cup \{a\}) - H(A_2)) \geq 0$, which completes the proof of submodularity of the weighted coverage term.

A.2 Proof of Submodularity of the Single-layer Diversity Term D(A)

Recall our definition of the single-layer diversity term $D(A)$.

$$\begin{aligned} D(A) &= \sum_{l=1}^{L} D_l(A) \\ &= \sum_{t,l} \sqrt{\sum_{j \in P_t^l \cap A} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} \end{aligned} \tag{A.2}$$

where P_t^l is the set of segments which belong to cluster t in layer l. V^l is the set of segments from layer l, and $|V^l|$ is the number of segments in layer l. w_{ij} measures the appearance similarity between vertices i and j. $\{P_t^l\}$ forms a partition of V^l, i.e. $P_t^l s$ are disjoint and $\cup_t P_t^l = V^l$.

Proof. From equation A.2, we have the marginal gain for element a given a selected set

A.

$$
\begin{aligned}
& D(A \cup \{a\}) - D(A) \\
= & \sum_{t,l} \sqrt{\sum_{j \in P_t^l \cap (A \cup \{a\})} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} - \sum_{t,l} \sqrt{\sum_{j \in P_t^l \cap A} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} \\
= & \sqrt{\sum_{j \in P_s^k \cap A} \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ij}) + \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ia})} - \sqrt{\sum_{j \in P_s^k \cap A} \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ij})}
\end{aligned}
$$

For $\forall\ A_1 \subseteq A_2 \subseteq V$ and any element $a \in V \setminus A_2$ (without loss of generosity, we can assume $a \in P_s^k$), we can compute:

$$
\begin{aligned}
& (D(A_1 \cup \{a\}) - D(A_1)) - (D(A_2 \cup \{a\}) - D(A_2)) \\
= & \left(\sqrt{\sum_{j \in P_s^k \cap A_1} \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ij}) + \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ia})} - \sqrt{\sum_{j \in P_s^k \cap A_1} \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ij})} \right) \\
& - \left(\sqrt{\sum_{j \in P_s^k \cap A_2} \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ij}) + \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ia})} - \sqrt{\sum_{j \in P_s^k \cap A_2} \frac{1}{|V^k|} (\sum_{i \in V^k} w_{ij})} \right) \\
= & \frac{(\sum_{i \in V^k} w_{ia}) / \sqrt{|V^k|}}{\sqrt{\sum_{j \in P_s^k \cap (A_1 \cup \{a\})} (\sum_{i \in V^k} w_{ij})} + \sqrt{\sum_{j \in P_s^k \cap A_1} (\sum_{i \in V^k} w_{ij})}} \\
& - \frac{(\sum_{i \in V^k} w_{ia}) / \sqrt{|V^k|}}{\sqrt{\sum_{j \in P_s^k \cap (A_2 \cup \{a\})} (\sum_{i \in V^k} w_{ij})} + \sqrt{\sum_{j \in P_s^k \cap A_2} (\sum_{i \in V^k} w_{ij})}}
\end{aligned}
$$

Because $A_1 \subseteq A_2$, we have $\sqrt{\sum_{j \in P_s^k \cap (A_1 \cup \{a\})} (\sum_{i \in V^k} w_{ij})} \leq \sqrt{\sum_{j \in P_s^k \cap (A_2 \cup \{a\})} (\sum_{i \in V^k} w_{ij})}$ and $\sqrt{\sum_{j \in P_s^k \cap A_1} (\sum_{i \in V^k} w_{ij})} \leq \sqrt{\sum_{j \in P_s^k \cap A_2} (\sum_{i \in V^k} w_{ij})}$. Therefore, we have $D(A_1 \cup \{a\}) - D(A_1)) - (D(A_2 \cup \{a\}) - D(A_2)) \geq 0$, which completes the proof of submodularity of the single-layer diversity term.

A.3 Proof of Submodularity of the Multi-scale Reward Term R(A)

Recall our definition of the multi-scale reward term $R(A)$.

$$
R(A) = \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap A} r_j} \tag{A.3}
$$

where V^l is the set of segments from layer l. The value r_j estimates the likelihood of a segment to be an object.

Proof. From equation A.3, we have the marginal gain for element a given a selected set A.

$$\begin{aligned} & R(A \cup \{a\}) - R(A) \\ = & \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap (A \cup \{a\})} r_j} - \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap A} r_j} \\ = & \sqrt{\sum_{j \in V^k \bigcap A} r_j + r_a} - \sqrt{\sum_{j \in V^k \bigcap A} r_j} \end{aligned}$$

For $\forall\ A_1 \subseteq A_2 \subseteq V$ and any element $a \in V \setminus A_2$ (without loss of generosity, we can assume $a \in V^k$), we can compute:

$$\begin{aligned} & (R(A_1 \cup \{a\}) - R(A_1)) - (R(A_2 \cup \{a\}) - R(A_2)) \\ = & \left(\sqrt{\sum_{j \in V^k \bigcap A_1} r_j + r_a} - \sqrt{\sum_{j \in V^k \bigcap A_1} r_j}\right) - \left(\sqrt{\sum_{j \in V^k \bigcap A_2} r_j + r_a} - \sqrt{\sum_{j \in V^k \bigcap A_2} r_j}\right) \\ = & \frac{\sum_{j \in V^k \bigcap A_1} r_j + r_a - \sum_{j \in V^k \bigcap A_1} r_j}{\sqrt{\sum_{j \in V^k \bigcap A_1} r_j + r_a} + \sqrt{\sum_{j \in V^k \bigcap A_1} r_j}} - \frac{\sum_{j \in V^k \bigcap A_2} r_j + r_a - \sum_{j \in V^k \bigcap A_2} r_j}{\sqrt{\sum_{j \in V^k \bigcap A_2} r_j + r_a} + \sqrt{\sum_{j \in V^k \bigcap A_2} r_j}} \\ = & \frac{r_a}{\sqrt{\sum_{j \in V^k \bigcap A_1} r_j + r_a} + \sqrt{\sum_{j \in V^k \bigcap A_1} r_j}} - \frac{r_a}{\sqrt{\sum_{j \in V^k \bigcap A_2} r_j + r_a} + \sqrt{\sum_{j \in V^k \bigcap A_2} r_j}} \end{aligned}$$

Because $A_1 \subseteq A_2$, we have $\sqrt{\sum_{j \in V^k \bigcap A_1} r_j} \leq \sqrt{\sum_{j \in V^k \bigcap A_2} r_j}$. Therefore, $(R(A_1 \cup \{a\}) - R(A_1)) - (R(A_2 \cup \{a\}) - R(A_2)) \geq 0$, which completes the proof of submodularity of the multi-scale reward term.

A.4 Proof of Submodularity of the objective function

Recall our objective function.

$$\begin{aligned} F(A) &= H(A) + \alpha D(A) + \beta R(A) \\ &= \sum_{i \in V} \max_{j \in A} w_{ij} - \sum_{j \in A} \phi_j + \alpha \sum_{n,l} \sqrt{\sum_{j \in P_t^l \cap A} \frac{1}{|V^l|} (\sum_{i \in V^l} w_{ij})} \\ &\quad + \beta \sum_{l=1}^{L} \sqrt{\sum_{j \in V^l \bigcap A} r_j} \end{aligned} \tag{A.4}$$

Proof. Based on the definition above, we can rewrite $(F(A_1 \cup \{a\}) - F(A_1)) - (F(A_2 \cup \{a\}) - F(A_2))$ as follows:

$$\begin{aligned} & (F(A_1 \cup \{a\}) - F(A_1)) - (F(A_2 \cup \{a\}) - F(A_2)) \\ = \; & \big(H(A_1 \cup \{a\}) - H(A_1)) - (H(A_2 \cup \{a\}) - H(A_2)\big) \\ & + \alpha \big(D(A_1 \cup \{a\}) - D(A_1)) - (D(A_2 \cup \{a\}) - D(A_2)\big) \\ & + \beta \big(R(A_1 \cup \{a\}) - R(A_1)) - (R(A_2 \cup \{a\}) - R(A_2)\big) \end{aligned}$$

Since α, β are non-negative, we have proved $(H(A_1 \cup \{a\}) - H(A_1)) - (H(A_2 \cup \{a\}) - H(A_2)) \geq 0$, $D(A_1 \cup \{a\}) - D(A_1)) - (D(A_2 \cup \{a\}) - D(A_2)) \geq 0$, and $(R(A_1 \cup \{a\}) - R(A_1)) - (R(A_2 \cup \{a\}) - R(A_2)) \geq 0$ in the previous sections. Therefore, we can have $F(A_1 \cup \{a\}) - F(A_1) \geq F(A_2 \cup \{a\}) - F(A_2)$, $\forall\ A_1 \subseteq A_2 \subseteq V$ and $a \in V \backslash A_2$, which completes the proof of submodularity property of the objective function $F(A)$.

Bibliography

[1] J. Carreira and C. Sminchisescu. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *PAMI*, 34(7):1312–1328, 2012.

[2] I. Endres and D. Hoiem. Category-independent object proposals with diverse ranking. *PAMI*, 36(2):222–234, 2014.

[3] P. Arbelaez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping, 2014. CVPR.

[4] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

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

[6] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9):1627–1645, 2010.

[7] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object recognition, 2013. IJCV.

[8] M. Cheng, Z. Zhang, W. Lin, and P. torr. Bing: Binarized normed gradients for objectness estimation at 300fps, 2014. CVPR.

[9] H. Teuber. Physiological psychology. *Annual review of psychology*, 6(1):267–296, 1955.

[10] J.M. Wolfe and T.S. Horowitz. What attributes guide the deployment of visual attnetion and how do they do it? *Nature Reviews Neuroscience*, pages 5:1–7, 2004.

[11] R. Desimone and J. Duncan. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 1995.

[12] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurbiology*, pages 4Nature Reviews Neuroscience:219–227, 1985.

[13] P. Arbelaez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation using regions and parts, 2012. CVPR.

[14] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling, 2012. ECCV.

[15] A. Y. S. Fidler, R. Mottaghi, and R. Urtasun. Bottom-up segmentation for top-down detection, 2013. CVPR.

[16] C.L. Zitnick and P. Dollar. Edge boxes: Locating object proposals from edges, 2014. ECCV.

[17] B. Singh, X. Han, Z. Wu, and L. Davis. Pspgc: Part-based seeds for parametric graph-cuts, 2014. ACCV.

[18] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *PAMI*, 34:2189–2202, 2012.

[19] A. Humayun, F. Li, and J. M. Rehg. Rigor: Reusing inference in graph cuts for generating object regions, 2014. CVPR.

[20] Tomasz Malisiewicz and Alexei A. Efros. Improving spatial support for objects via multiple segmentations. In *British Machine Vision Conference (BMVC)*, September 2007.

[21] X. Chen, A. Jain, A. Gupta, and L. Davis. Jigsaw puzzle: Piecing together the segmentation jigsaw using context, 2011. CVPR.

[22] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik. Simultaneous detection and segmentation, 2014. ECCV.

[23] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks, 2007. KDD.

[24] M. Liu, R. Chellappa, O. Tuzel, and S. Ramalingam. Entropy-rate clustering: Clustering analysis via maximizing a submodular function subject to a matroid constraint. *PAMI*, 36(1):99–112, 2013.

[25] G. Kim, E. Xing, L. FeiFei, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion, 2012. CVPR.

[26] Z. Jiang and L. Davis. Submodular salient region detection, 2013. CVPR.

[27] R. Liu, Z. Lin, and S. Shan. Adaptive partial differential equation learning for visual saliency detection, 2014. CVPR.

[28] F. Zhu, Z. Jiang, and L. Shao. Submodular object recognition, 2014. CVPR.

[29] G. Nemhauser, L. Wolsey, and M. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.

[30] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[31] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering, 2004. NIPS.

[32] R. D. Galvao. Uncapacitated facility location problems: contributions. *Pesquisa Operacional*, 24(1):7–38, 2004.

[33] D. Martin, C. Fowlkes, and J. Malik. Learning to find brightness and texture boundaries in natural images, 2002. NIPS.

[34] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[35] J. Kim and K. Grauman. Shape sharing for object segmentation, 2012. ECCV.

[36] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders. Segmentation as selective search for object recognition, 2011. ICCV.

[37] H. Lin and J. Bilmes. Learning mixtures of submodular shells with application to documnet summarization, 2012.

[38] M. Gygli, H. Grabner, and L. V. Gool. Video summarization by learning submodular mixtures of objectives, 2015. CVPR.

[39] M. Yang and L. Zhang. Gabor feature based sparse representation for face recognition with gabor occlusion dictionary, 2010. *ECCV*.

[40] J. Wright, A. Ganesh, S. Rao, and Y. Ma. Robust face recognition via sparse representation. *TPAMI*, pages 31(2):210–227, 2009.

[41] D. Bradley and J. Bagnell. Differential sparse coding, 2008. *NIPS*.

[42] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, pages 54(1):4311–4322, 2006.

[43] J. Mairal, F. Bach, J. Ponce, G. Saprio, and A. Zisserman. Discriminative learned dictionaries for local image analysis, 2008. *CVPR*.

[44] D. Pham and S. Venkatesh. Joint learning and dictionary construction for pattern recognition, 2008. *CVPR*.

[45] Xiaohui Shen and Ying Wu. A unified approach to salient object detection via low rank matrix recovery, 2012. *CVPR*.

[46] Z. Zhang, Y. Matsushita, and Y. Ma. Camera calibration with lens distortion from low-rank textures, 2011. *CVPR*.

[47] J. Lee, B. Shi, Y. Matsushita, I. Kweon, and K. Ikeuchi. Radiometric calibration by transform invariant low-rank structure, 2011. *CVPR*.

[48] B. Cheng, G. Liu, J. Wang, Z. Huang, and S. Yan. Multi-task low-rank affinities pursuit for image segmentation, 2011. *ICCV*.

[49] X. Cui, J. Huang, S. Zhang, and D. Metaxas. Background subtraction using group sparsity and low rank constraint, 2012. *ECCV*.

[50] T. Zhang, B. Ghanem, and N. Ahuja. Low-rank sparse learning for robust visual tracking, 2012. *ECCV*.

[51] C. Zhang, J. Liu, Q. Tian, C. Xu, H. Lu, and S. Ma. Image classification by non-negative sparse coding, low-rank and sparse decomposition, 2011. *CVPR*.

[52] C. Chen, C. Wei, and Y. Wang. Low-rank matrix recovery with structural incoherence for robust face recognition, 2012. *CVPR*.

[53] L. Ma, C. Wang, B. Xiao, and W. Zhou. Sparse representation for face recognition based on discriminative low-rank dictionary learning, 2012. *CVPR*.

[54] Zhuolin Jiang, Zhe Lin, and Larry S. Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd, 2011. *CVPR*.

[55] Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition, 2010. *CVPR*.

[56] J. Yang, K. Yu, and T. Huang. Supervised translation-invariant sparse coding, 2010. *CVPR*.

[57] J. Mairal, F. Bach, J. Ponce, G. Saprio, and A. Zisserman. Supervised dictionary learning, 2009. *NIPS*.

[58] J. Wang, A. Yang, K. Yu, F. LV, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification, 2010. *CVPR*.

[59] J. Wright, A.Ganesh, S. Rao, Y. Peng, and Y. Ma. Robust pricipal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. *submitted to Journal of the ACM*, 2009.

[60] M. Yang, L. Zhang, J. Yang, and D. Zhang. Robust sparse coding for face recognition, 2011. *CVPR*.

[61] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, pages 15(1)2):3736–3745, 2006.

[62] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. *IEEE Transactions on Information Theory*, 2008.

[63] Z. Lin, M. Chen, and Y. Ma. The argumented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *UIUC Tech. Rep. UIUC-ENG-09-2214*, 2011.

[64] E. Candès, X. Li, Y. Ma, and J. Wright. Robust pricipal component analysis? *Journal of the ACM*, 58, 2011.

[65] G. Liu, Z. Liu, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[66] R. Basri and D. Jacobs. Lambertian reflectance and linear subspaces, 2001. *ICCV*.

[67] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penality for low rank representation, 2011. *NIPS*.

[68] L. Zhuang, H. Gao, Z. Lin, Y. Ma, X. Zhang, and N. Yu. Non-negative low rank and sparse graph for semi-supervised learning, 2012. *CVPR*.

[69] G. Golub, P. Hansen, and D. O'leary. Tikhonov regularization and total least squares. *SIM J.Matri Anal. Appl.*, pages 21(1):185–184, 1999.

[70] G. Zhang, Z. Jiang, and L. Davis. Online semi-supervised discriminative dictionary learning for sparse representation, 2012. *ACCV*.

[71] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *TPAMI*, pages 23(6):643–660, 2001.

[72] A. Martinez and R. Benavente. The ar face database, 1998. *CVC Technical Report 24*.

[73] Fei-Fei. Li, R. Fergus, and P. Perona. Learning generative visual models from few training samples: An incremental bayesian approach tested on 101 object categories, 2004. *CVPR Workshop on Generative Model Based Vision*.

[74] Y. Zhang, Z. Jiang, and L. Davis. Learning structured low-rank representations for image classification, 2013. *CVPR*.

[75] T. Hazan, S. Polak, and A. Shashua. Sparse image coding using a 3d non-negative tensor factorization, 2005. *ICCV*.

[76] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos. Tensor sparse coding for region covariances, 2010. *ECCV*.

[77] K. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kerne-based learning algorithms. *IEEE Transactions on Neural Networks*, (2):181–201, 2001.

[78] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification, 2006. *ECCV*.

[79] O. Tuzel, F. Porikli, and P. Meer. Human detectio via classification on riemannian manifolds, 2007. *CVPR*.

[80] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos. Positive definite dictionary learning for region covariances, 2011. *ICCV*.

[81] M. Harandi, C. Sanderson, R. Hartley, and B. Lovell. Sparse coding and dictionary learning for symmetric positive definite matrices: A kernel approach, 2006. *ECCV*.

[82] C. Yuan, W. Hu, X. Li, S. Maybank, and G. Luo. Human action recognition under log-euclidean riemannian metric, 2009. *ACCV*.

[83] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on riemannian manifolds. *PAMI*, 30(10):1713–1727, 2008.

[84] J. Davis, B. Kulis, P. Jain, S. Sra, and I. Dhillon. Information-theoretic metric learning, 2007. *ICML*.

[85] M. Grant and S. Boyd. Cvx: Matlab software for disciplined convex programming, 2010. http://cvxr.com/cvx.

[86] I. Ramirez, P. Sprechmann, and G. Sapiro. Classification and clustering via dictionary learning with structural incoherence and shared features, 2010. *CVPR*.

[87] T. Randen and J. Husøy. Filtering for texture classification: a comparative study. *IEEE Trans. Pattern Anal. Mach. Tntell.*, pages 291–310, 1999.

[88] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, ISBN: 0387952845, 2003.

[89] T. Lee. Image representation using 2d gabor wavelets. *PAMI*, pages 959–971, 1996.

[90] Y. Cong, J. Yuan, and J. Liu. Sparse reconstruction cost for abnormal event detection. In *CVPR*. 2011.

[91] J. Kim and K. Grauman. Observe locally, infer globally: A space-time mrf for detecting abnormal activities with incremental updats. In *CVPR*. 2009.

[92] B. Zhao, L. Fei-Fei, and E. Xing. Online detection of unusual events in videos via dynamic sparse coding. In *CVPR*. 2011.

[93] B. Kedem and S. Wen. Semi-parametric cluster detection. *Journal of Statistical theory and Practice*, 1(1):49–72, 2007.

[94] X. Cui, Q. Liu, M. Gao, and D. Metaxas. Abnormal detection using interaction energy potentials, 2011.

[95] R. Mehran, B. E. Moore, and M. Shah. A streakiline representation of flow in crowded scenes. In *ECCV*. 2010.

[96] T. Xiang and S. Gong. Video behavior profiling for anomaly detection. *PAMI*, 30:893–908, 2008.

[97] M. Bertini, A. D. Bimbo, and L. Seidenari. Multi-scale and real-time non-parametric approach for anomaly detection and localization. *CVIU*, 116:320–329, 2012.

[98] O. Boiman and M. Irani. Detecting irregularities in images and in video. In *ICCV*. 2005.

[99] R. Mehran, B. E. Moore, and M. Shah. Abnormal crowd behavior detection using social force model. In *CVPR*. 2009.

[100] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video, 204. CVPR.

[101] S. Wu, B. E. Moore, and M. Shal. Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes, 2010. CVPR.

[102] V. Mahadevan, W. Li, B. Bhalodia, and N. Vasoncelos. Anomaly detection in crowded scenes, 2012. CVPR.

[103] J. Qin and B. Zhang. A goodness-of-fit test for logistic regression models based on case-control data. *Biometrika*, 84(3):609–618, 1997.

[104] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz. Robust real-time unusual event detectio nusing multiple fixed-location monitors. *PAMI*, 30:555–560, 2008.

[105] C. Liu. Beyond pixels: Exploring new representations and applications for motion analysis, 2009. Doctoral Thesis.