Dictionary learning and sparse representation for image analysis with application to segmentation, classification and event detection

---

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy

by

Rituparna Sarkar

May 2017

Approval Sheet

This dissertation

is submitted in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Rituparna Sarkar

This dissertation has been read and approved by the Examining Committee:

Scott T. Acton, Advisor

Zongli Lin, Committee Chair

Daniel Weller

Kevin Skadron

Laura Barnes

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, Dean, School of Engineering and Applied Science

May 2017

Abstract

Sparse representation based dictionary learning has been exploited in solving various image analysis problems - image classification, tracking, quality assessment, de-noising, image reconstruction. The objective of dictionary learning is to obtain an adaptive basis function from the data and simultaneously provide a compact representation. In this work we employ sparse representation based dictionary learning techniques for segmentation, image classification and video analysis problems.

In image and video processing applications, one of the major challenges is the choice of appropriate features for image representation. Various techniques exist that employ different analytical methods to extract color, texture and frequency information from images. However, these methods do not identify which of this information are more relevant for a particular image. Neither do these methods have any discrimination power to recognize more informative local image regions.

In this work, we first tackle the problem of query specific image feature descriptor selection. Depending on the image content, different features e.g., color texture, structure can prove to be more relevant in representing and discriminating an image. We use a discriminative dictionary learning method in designing a classifier and an information theoretic measure to select the most appropriate feature for an image. This method attempt to identify the feature descriptors that provide more information about an image conditioned on the available images in a class.

In image classification, while identifying the relevant feature type is important, it is also

crucial to identify the essential contents of an image which discriminate it from the others. While the above mentioned solution is appropriate for determining image specific feature type, it does not incorporate any local image analysis to identify image regions associated with an object. To address this problem, we develop a method that leverages salient object detection framework to learn the dictionary and sparse codes from an image. The method simultaneously detects relevant image regions and computes a compact image representation. We also devise similarity measures exploiting the sparse representations for comparing image pairs. This similarity measure is used in image classification particularly for scenarios where training data is limited. Our method outperformed the state of the art methods by an average of 12% in overall accuracy for histo-pathological tissue image classification

Although the above mentioned saliency guided dictionary learning method is applied to image classification, the application is not limited to just object recognition. The method is hence exploited for event detection from video. The saliency based dictionary learning and the similarity measure is used first for a frame by frame analysis to identify the temporal occurrence of an event. To make the system more robust to occlusion, dynamic background, we further employ a spatio-temporal saliency driven low rank and sparse representation scheme. The technique reconstructs the salient regions as foreground and low saliency regions as background. The methods were validated for applications of unusual and hazardous event detection from videos and achieved significant improvement over state of the art background subtraction methods for anomaly detection.

Acknowledgments

I would like to thank my adviser, Dr. Acton for giving me the opportunity to work under him as a graduate student. I consider myself extremely lucky to have Scott as my advisor who has been extremely supportive and helped me overcome the challenges in academic and personal fronts during my years as a graduate student at University of Virginia. I am grateful to my committee members, Dr. Skadron, Dr. Lin, Dr. Weller and Dr. Barnes, for their support and encouragement. I would also like to thank my Master's advisor Dr. Namrata Vaswani from Iowa State University for her help and support during my initial years in USA.

I am thankful to all my colleagues at VIVA. I would definitely miss the extended lunch and the coffee sessions at the Corner and SJ. Thanks to all my friends for staying by my side during the difficult times and help me stay focused on my goal. Thank you Kakali Bhattacharya for all your support and suggestions. It is always good to know you have someone to reach out to for anything and everything.

I would not have been able to accomplish my goals without the support of Maa and Baba. Words are not enough to express my gratitude towards you but thank you for being by my side and supporting all my decisions.

Finally, I would like to acknowledge my husband and fellow graduate student Suvadip Mukherjee. Thank you for all your help, support, encouragement and not to forget your criticisms. Thanks for being by my side.

Contents

List of Figures

List of Tables

List of Symbols

Symbol	Description
y	lower case variable denotes a scalar
$\mathbf{y}$	Bold faced lower case variable denotes a vector
$\mathbf{Y}$	Bold faced upper case variable denotes a matrix
$\mathbf{y_i}$	Bold faced variable with subscript i denote a column in a matrix
$\mathbf{y^i}$	Bold faced variable with superscript i denote a row in a matrix
$\mathbb{R}^n$	Real coordinate space f dimension n
$\|.\|_0$	ℓ_0 norm i.e., number of non zero elements in a vector
$\|.\|_1$	ℓ_1 norm i.e., sum of absolute values of the vector elements
$\|.\|_2$	ℓ_2 norm i.e., square root of the sum of absolute value squares of vector elements
$\|.\|_F$	Frobenius norm i.e., square root of the sum of squares of absolute values of matrix elements
$(.)^T$	Transpose operator
$(.)^{-1}$	Matrix inverse operator
$\mathbf{D}$	Dictionary or over-complete basis. Each column $\mathbf{d}_i$ is called *atom*
$\mathbf{X}$	Sparse representation of data. Each column $\mathbf{x}_i$ is a sparse vector.
$\frac{\partial(.)}{\partial t}$	Partial derivative with respect to variable t
Ω	Image domain
ϕ	Level set function
$H_\epsilon(\phi)$	Regularized Heaviside function

$\delta_\epsilon(\phi)$	Regularized delta function
∇	Gradient operator
$\mathscr{H}(.\|.)$	Conditional entropy
$\mathbb{I}$	Identity matrix
$KL(p\|\|q)$	KullbackLeibler divergence between probability distributions p and q
$\tilde{D}$	Degree matrix of a graph
$\mathbb{N}_i$	Neighborhood of i
$\mathcal{E}(i,j)$	Edge between two nodes i and j in a graph
Ψ	Non-linear transform from feature space to kernel feature space
$\mathscr{K}(.)$	Kernel function, where $\mathscr{K}(.) = \hat{\phi}(.)^T\hat{\phi}(.)$.
$\mathcal{K}$	Linear combination of different kernel functions $\mathscr{K}(.)$
$\hat{\phi}$	Non linear transform of feature space to a higher dimensional space
$\mathbb{D}$	Dictionary learned from Kernel representation of features.

Chapter 1

Introduction

During the past decades there have been numerous attempts to simulate the working principles of the human brain in signal and image processing. A major challenge in this area is to perceive how a natural scene is processed in the human visual cortex [3], which aids in distinguishing different patterns in the signals. Olshaussen and Field [4, 5] proposed a theory in this context which states that neurons in human visual cortex can be characterized by spatially localized, oriented, bandpass filters. Additionally during the processing of sensory information, only a few of these neurons are activated. Such processing mechanism has multiple advantages: energy and storage efficiency, the elimination of redundancy and extraction of explicit information from natural signals. This nature of human sensory neurons motivated the sparse coding technique in processing natural images, where an image is modeled as a linear combination of basis functions. The coefficients of the linear combination are characteristic image representation which are unique for a particular image. The sparse codes for image and a basis function which mimic the neurons in human visual cortex can be computed using an efficient coding scheme.

Sparse coding techniques usually involve representing an image as a linear combination of an over-complete basis where only a few of the basis elements are used in representation. Thus sparse coding techniques achieve two fold advantage: (i) it projects the inherent patterns

to a higher dimensional space, helping the creation of more unique representations, (ii) the sparse code are generated using only a few of the basis elements, that reduces the size of the representations facilitating significant compression of signals, storage and transmission. To achieve an efficient sparse coding, the basis used for the representation need to be designed appropriately. Off-the-shelf basis functions such as wavelets, Fourier, have been used extensively in traditional image processing applications. These basis functions are efficient in particular low level images analysis applications such as de-noising (Fourier), compression (DCT, wavelet). While these pre-defined basis functions have been successfully applied in certain applications, in recent years researchers have focused on obtaining data specific dictionary [6–9], that would provide more compact and unique representations.

1.1 Scope of the thesis

Sparse representation based dictionary learning has been an active area of research in the field of image processing and computer vision with applications ranging from, image retrieval [10, 11], classification [9, 12–16], segmentation [17, 18] to video tracking [19, 20], scene change detection [21]. Sparse coding techniques provide compact representations but do not incorporate discrimination. Hence consolidating discriminatory functions or selecting more distinguishing characteristics from the data can help to adequately exploit the sparse coding scheme.

A major challenge in image analysis applications lies with the selection of relevant image features. Various analytical methods exist in literature [22–24] which extract image information such as color, frequency, texture, commonly called *feature descriptor*. However, the content variation in images implies that just one of these feature descriptors cannot appropriately describe every image. Accordingly, one needs to identify the more suitable and discriminative feature descriptor congruent with the image content. In high level computer vision applications, while identifying the ideal feature descriptor is necessary, it is also critical

to recognize local image regions which are responsible for differentiating between images, or in other words specify the subject of an image.

In this thesis we aim at **integrating a discriminative feature selection problem in conjunction with learning an over-complete basis for a sparse representation**. This achieves two-fold advantage: (a) identifying discriminative features, (b) extracting compact patterns exploiting the selected features. The generated sparse codes are used in the application of image classification. We further demonstrate that the developed method is not limited to the application of image recognition and a similar approach can be exploited in event detection from video. In this section we detail the existing challenges related to the selection of relevant feature for compact image representation. In the following part we discuss the challenges specific to the applications of segmentation, image classification and event detection from video.

Image segmentation: Image segmentation methods address the problem of automatically detecting region(s) of interest from an image. They are typically used to spatially locate objects or extract boundaries of objects for further investigation. In biomedical image analysis, segmentation is commonly used for detection of tumors, cells and cellular components to help in the diagnosis, analyzing progression of the diseases, surgery planning or simply for behavioral study in presence/absence of certain stimuli. In non-biological application segmentation is typically used for applications of detection of faces, objects which can further be used for recognition and tracking.

Generally, in applications involving segmentation, manually detecting objects or tracing their boundaries are time consuming, hence the need for automatizing the process. However, given the variety of imaging techniques, developing a generic segmentation method is impractical. In this thesis we mainly address the problem of ultrasound image segmentation in presence of substantial amount of noise, clutter and illumination variation.

- *Segmentation in presence of inhomogeneous intensity*: Although in literature, there exists a considerable amount of research on segmentation exists, modeling the intensity variation in images is still an open ended problem. In presence of noise, clutter and intensity variations, good quality edge information is not available. In such scenarios, some of the traditional segmentation models [25, 26], where the curve propagation is dominated by intrinsic image properties such as edge, intensity are not effective. Hence there is the need to resort to region base segmentation techniques where the image can be partitioned into regions exhibiting similar properties e.g., color, texture, *etc.* However, in presence of varying intensity inside a particular object, a more sophisticated method which can model the intensity variation is desired. To deal with this we employ a data driven method to learn the intensity variation pattern present in a particular application of ultrasound imaging technique.

Image classification: Image retrieval or classification techniques aim at identifying similar images from a pre-labeled dataset. For classification purpose, initially a set of image data is collected and each of the images is annotated by a category consistent with the image content. The objective is to automatically decide the category of any new test image based on the model learned from the training dataset. For example, an interesting application is biometric classification, which involves identification of persons based on images of fingerprints, iris, face and others. Image recognition methods are popular in biomedical imaging as well, and are used for various purposes ranging from shape based cell categorization, assessment of tumors, detection of diseases from histo-pathological images, *etc.*. Security, surveillance and industrial automation are other areas which have extensive use of image retrieval and classification.

Image classification involve various approaches to identify images or parts of images based on a given image database. Generalized image classification faces numerous challenges. One such challenge is the choice of image features to appropriately describe an object. Identifying the image features that are characteristic of a particular object and simultaneously discriminate

between different objects is critical. Obtaining meaningful features from an image or a particular image category is desired in computer vision applications. Feature selection can occur at two levels in image classification: identifying either a feature descriptor for an image or relevant local regions which contain object level information.

- *Feature descriptor selection*: This denotes the selection of feature types from a larger pool of different feature descriptors. The feature descriptors typically capture the color, texture and frequency content of an image, which are computed using some analytical formulation. Commonly used feature extraction methods, [24,27–29] encode the frequency, texture or color content of an image. The scale-invariant feature transform SIFT [22], for example, attempts to identify object key-points and extract features surrounding the points. While [23] encodes the local gradients of an image to extract structural information. However, a single feature descriptor may not be sufficient to represent all images universally. For example, images of flags of different countries may be differentiated by color features, while structural features may be necessary in discriminating between buildings. In most cases of image classification, the object to be retrieved is unknown and the image database consists of various object categories. Hence the type of feature that would be more discriminative for the recognition of a particular object is not known. In such scenarios, an adaptive feature selection or fusion approach is necessary for the discriminative representation of an object.
- *Leveraging relevant local features*: While high level feature selection methods adaptively select feature descriptors, these methods do not differentiate objects from background, or discriminate if a local feature is more significant in representing an image. The major challenge in this respect to identify the object or the local features of interest from the image. Most of the object recognition approaches are based on global image information. Local features are combined in was that yield global information of the image. Recognition task is then performed exploiting these features. On the other hand, if the object of interest can be extracted from the image, more essential and object related characteristics

can be extracted. Meaningful features often relate to the discriminative feature descriptors from the image that capture object level information. Challenges still remain in the field of detecting objects in an image, which are considered salient by the human vision system and how to leverage this saliency for object recognition problems.

Event detection: Applications of event detection denote detecting the temporal occurrence of anomalies which differ significantly from regular patterns. Pertaining to surveillance and traffic videos, an interesting problem is to recognize any unwanted incidents e.g., accidents, collisions, sudden fire, *etc.* In addition to recognizing the occurrence of such incidents, detecting the time of the incident is also of critical importance. Sparse and low rank representation of a temporal sequence is commonly used in anomaly or event detection. In these methods, a sequence is represented as superposition of low rank and sparse matrix, where the low rank represents the background and the sparse matrix gives a notion of the event. But in videos with highly dynamic background identifying events from the sequence using the sparse and low rank model is still a challenging task.

- *Leveraging local spatio-temporal feature*: While selection of relevant features aids in image recognition problems, it can also be employed in analyzing and detecting sudden incidents taking place in videos. Salient object detection, or saliency map computation mimics the human vision by determining regions in an image that draw human attention. Temporal analysis of detected salient regions or computation of simultaneous spatio-temporal saliency map can be exploited in solving this problem. In this thesis, we envision a spatio-temporal saliency map to identify potential event regions. This aids in addressing the challenges posed by the low rank, sparse representation techniques for event detection from videos with dynamic background.

1.2 Objectives and Contribution

The objective of this thesis is to exploit the basics of low-level image analysis techniques in applications such as object detection and subsequently classification. Given the scope of the thesis the goal can be consolidated as the solution of three main questions in image analysis and computer vision applications:

i. What are the more discriminative features for an image which can describe and differentiate the contents of images more accurately?

ii. How and in what ways can feature selection enhance and benefit the sparse coding technique for image classification?

iii. Are the benefits only limited to classification?

To address the above challenges, we envision a saliency detection engine that will aid in useful feature detection and an integrated sparse representation framework for classification and recognition of images. We then extend the approach to a temporal sequence of images to detect unusual and hazardous events from videos. The main emphasis of this thesis is to make advances related to dictionary learning paradigm integrated with discriminative feature selection. In this section we state the specific problems, and our approach to solve them.

Contribution 1: The salient idea of our **first** contribution is to compute the optimal set of functions to model the region intensities. This provides an elegant solution to deal with intensity inhomogeneities prevalent in many imaging applications such as ultrasound and fluorescence microscopy. In DL2S [17], we employ the method of dictionary learning to learn the basis functions for image representation. For a particular application, if data with varying intensity profile is available, we hypothesize that the learned basis function can capture the finer details in addition to the mean intensity of the images. To our knowledge, DL2S is the first of its kind lo learn a data adaptive dictionary to model image intensities for segmentation. We integrate the dictionary learning with the region based Chan-Vese [30]

model for ultrasound image segmentation. Our approach models the inside and outside region of an object as linear combination of the learned dictionary.

Contribution 2: The **second** objective is to solve the *feature descriptor selection* problem for image classification. In this work, **Meta-algorithm** for image feature nomination [12], we aim at developing an image classification framework based on dictionary learning and sparse representation of images. A discrimination function, integrated in the dictionary learning method, aids in differentiating features belonging to different classes. The feature descriptor selection is performed after the classification decision is made using the different feature types for the query image. The query image specific feature type nomination is based on information theoretic measures which provide higher mutual information between the query and the determined image category.

Contribution 3: The **third** contribution integrates a *low level feature selection* method with dictionary learning technique for image classification. The low level features are obtained by way salient region detection.

In the **second** problem, we design a method to leverage local relevant features from an image. Super-pixel based over-segmentation of the images is performed to extract features and determine the salient object regions in the image. These local features and the saliency are then adopted in a dictionary learning framework to obtain the sparse codes. We develop a saliency guided dictionary learning framework (**SDL**) where we learn the over-complete basis from an image while emphasizing the reconstruction of image patches based on their saliency values. Salient object detection is exploited here to provide information regarding the importance of the candidate features. We also propose an extended work where the salient object detection can be refined while updating the dictionary (**SDLs**) to adjust between reconstruction error and saliency values. In literature, methods have been proposed which threshold the saliency map to retain salient image features for different applications.

Our proposed method is the first of its kind to integrate the salient feature detection with sparse representation based dictionary learning for image classification.

Since in this work the dictionary is learned for each image in an unsupervised manner, a robust similarity measure is necessary to compare pair of images to perform image classification. The adoption of saliency in the sparse coding framework is particularly advantageous in devising the similarity measure. When comparing pair of images with similar content, the learned dictionary represents the discriminative image features with greater accuracy and yields approximately similar sparse codes. Consequently we employ the learned sparse representations to develop a similarity measure to compare images and exploit this for image classification in applications where training data is limited.

In this approach we develop similarity measures by comparing the compressibility of the sparse codes between a pair a of images when represented with each other's dictionary. Further, the contribution of each dictionary atom in representing the image is accounted for, in obtaining a histogram of the sparse codes. This sparse code histograms are employed in computing the similarity between images.

Contribution 4: As stated earlier, the use of saliency guided dictionary learning and the proposed similarity measure are not limited to image classification. In our **fourth** contribution we aim to detect and analyze unusual and anomalous events in a video. While detecting sudden incidents occurring in a video, often prior knowledge abut the incidents is not available. Hence supervised machine learning approaches are not appropriate in such scenarios. To solve this we exploit the saliency detection based dictionary learning framework and the similarity measure in a spatio-temporal $(2d + time)$ framework. The objective is to detect the spatial location of the change as well as to identify the time of occurrence. We first employ the method developed in **SDL** and exploit that for a frame by frame analysis of the video. We further refine the method to accommodate for a low rank representation to better handle the dynamic backgrounds.

1.3 Thesis outline

The thesis is arranged as follows. In Chapter 2, we provide the background and state of the art methods. The proposed methods involve three broad image analysis disciplines - dictionary learning, feature selection and similarity measure. Therefore, to present a review of the literature, we categorize the prior art in three distinct categories. Since the different applications are mainly based on dictionary learning techniques, we further provide background on sparse representation based dictionary learning in image processing problems of segmentation, classification and video analysis with applications to tracking and event detection.

In Chapter 3 we describe the dictionary learning level set **DL2S** for image segmentation in presence of intensity inhomogeneity and apply it for ultrasound image segmentation.

In Chapter 4 we describe the **Meta-algorithm** for feature nomination and demonstrate the efficacy of our proposed method in image classification.

In Chapter 5, we discuss our third contribution of salient feature guided dictionary learning (**SDL** and **SDLs**). Then we show two approaches of similarity evaluation exploiting the sparse representations. The method is used in classification of histo-pathological tissue images. We also show a second application of military vehicle recognition using the above mentioned approach.

In Chapter 6 we discuss the use of this saliency guided dictionary learning and the proposed similarity measures in detecting sudden anomalous events from video captured using car-mounted or hand held camera. We further show that the frame by frame analysis can be extended to a volumetric approach for identifying the temporal occurrence of the events.

In Chapter 7 we summarize the methods and discuss the possible future extensions and other applications of the developed methods.

Chapter 2

Background

Sparse representation based dictionary learning has been used in various applications of image processing and computer vision. The literature is vast and spans the problems of image denoising, registration, segmentation, classification, quality assessment and extends to analysis of video for object tracking, event detection, activity recognition, *etc.* [1, 6, 8, 14, 16, 19, 20, 31–37]. Since the works presented in this thesis is broadly related to three main topics, we divide the background in three different sections. In this chapter we first give a background on sparse representation based dictionary learning which forms the main basis of the works proposed in this thesis. The background on the three main applications of the thesis: segmentation, image classification and event detection using dictionary learning is also discussed here. Then in the next section background on different *feature type* and *object-relevant* local feature selection methods are discussed. Finally, we discuss the literature on similarity measures used in image analysis.

2.1 Dictionary learning

Sparse coding aims at representing a signal as a linear combination of basis function by exploiting the redundancy in an over-complete basis [38–40]. In image analysis applications, sparse coding techniques exploit the sparseness in natural image statistics. In linear represen-

tation of data, a given signal $\mathbf{y} \in \mathbb{R}^m$ is represented as a linear combination of some basis function given as,

$$\mathbf{y} \approx \mathbf{D}\mathbf{x} \tag{2.1}$$

Here the columns of $\mathbf{D}$ act as basis functions and $\mathbf{x}$ contains the coefficients of linear combination. Sparse coding techniques resort to the fact that only a few of the columns actually contribute in reconstructing the signal i.e., most of the elements in $\mathbf{x}$ are zero. Given a basis function $\mathbf{D} \in \mathbb{R}^{m \times K}$, the optimization for sparse representation is given as

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 \leq \epsilon \tag{2.2}$$

$\mathbf{x} \in \mathbb{R}^K$ is the corresponding sparse representation of the data $\mathbf{y}$. The ℓ_0 norm imposes the sparsity constraint on $\mathbf{x}$.

Traditional image analysis applications generally use pre-selected basis functions e.g., Fourier, wavelet, also called as analytic basis functions (since they are derived from pre-defined mathematical formulation). In applications like image de-noising (Fourier, wavelet), image compression (DCT), these analytic dictionaries have shown promising result. However, each of these pre-defined basis are reasonable in representing different types of signals [41]. For e.g., Fourier basis is good for representing more uniformly smooth signals. Discontinuities in signals generate high coefficients over all the frequencies and thus imposing sparsity constraint lead to higher reconstruction error. On the other hand the wavelet basis is good for approximating point singularities, but in images, where edges are more prominent, wavelet is not a good choice for representation. These analytic dictionaries are easy to implement but are only appropriate in representing particular type of data.

In recent years research focus has been on developing data driven basis functions. A well known and extensively used method in this context is *Principle Component Analysis (PCA)* [42], where the basis functions are learned form the data. Principle Component

Analysis aims at finding a new orthogonal coordinate system. The co-ordinates are ordered in a way that the first coordinate principle component (PC) is in the direction of maximum variability in the data, second PC is in the direction of the second highest variation and so on. Once the PCs are obtained the data can now be projected on this new coordinate system which gives maximum separation between different dimensions. However, not all the PCs are required to represent the data, only the first few can be used thus significantly reducing the data dimension. For scenarios where the data distribution cannot be characterized by a mean and covariance, PCA cannot perform efficiently. A more generalized methods is *Independent component analysis (ICA)* [43]. Here the goal is to obtain a linear representation of non-Gaussian distributed data with assumption that the components are statistically independent. The method aims to find a transformation matrix such that the transformed data is sparse. However, the assumptions of statistical independence is not always true in practical scenarios.

Olshausen and Field [4], first demonstrated that an over-complete basis learned from the data (image patches in this case) for sparse representation mimics the human vision system. Instead of representing all data using a single set of learned basis as in PCA, the data can be represented as a linear combination of a subset of the over-complete basis. This concept of choosing a subset of the basis to represent the data introduces the idea of sparsity. Olshausen and Field [4] demonstrated that this theory is in unison with the working principle of the human visual system. Henceforth a number of work has been proposed in the literature to obtain a data driven basis function [6, 44], where the dictionary is learned from the available data for a more compact representation.

The main objective is to design an over-complete basis function from the available data (image or image patches) for a sparse compact representation. Given a set of images or image patches denoted by $\mathbf{y}_i$, where i is the total number of images (image patches), the dictionary

learning algorithm solves the following optimization,

$$\min_{\mathbf{D},\mathbf{x}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 \text{ s.t. } \|\mathbf{x}_i\|_0 \leq \tau, \quad \forall i \tag{2.3}$$

Here $\mathbf{D}$ is the dictionary learned and $\mathbf{x}_i$ the sparse representation for each data $\mathbf{y}_i$. $\|.\|_0$ is the ℓ_0 norm that denotes number of non zero elements in a vector and τ allowable maximum number of non-zero elements used for representation. Each column of the dictionary is often denoted as *atoms*.

The method of optimal directions (MOD) [44] solves the objective function in (2.3) by alternating between sparse coding and solving for the dictionary. The dictionary is solved in the least square sense, i.e., $\mathbf{D} = \mathbf{Y}\mathbf{X}^{\dagger}$, where $\mathbf{X}^{\dagger}$ is the pseudo inverse of $\mathbf{X}$.

In [6], the K-SVD, so far the most popular algorithm for dictionary learning was introduced. The K-SVD algorithm also solves the optimization in (2.3) as a two-step approach. First, the sparse solution is obtained by minimizing the following.

$$\min_{\mathbf{x}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 \quad \text{s.t.} \quad \|\mathbf{x}_i\|_0 \leq \tau, \quad \forall i \tag{2.4}$$

The orthogonal matching pursuit (OMP) algorithm [45] is used to solve (2.4). The next step updates the dictionary using the sparse codes obtained from (2.4). Instead of solving the dictionary as a least square solution as in MOD, the K-SVD updates each column of the dictionary, by solving a low rank approximation problem.

The dictionary is updated by solving $\min_{\mathbf{D}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2$. The objective function is re-written as

$$\sum_{i=1}^{N} \|y_i - D\mathbf{x}_i\|_2^2 \;=\; \|\mathbf{Y} - \sum_{j=1, j\neq k}^{N} \mathbf{D}_j \mathbf{X}^j - \mathbf{d}_k \mathbf{x}^k\|_2^2 \;=\; \|\mathbf{E}_k - \mathbf{d}_k \mathbf{X}^k\|_F^2 \tag{2.5}$$

Thus $\mathbf{d}_k$ is obtained by taking the singular value decomposition of $\mathbf{E}_k$=$\mathbf{U}\mathbf{\Sigma}\mathbf{V}$ and $\mathbf{d}_k = \mathbf{U}(:, 1)$. The singular value decomposition(SVD) is done K times which is the number of columns in

D in each iteration. A consolidated description of K-SVD is given in algorithm 1

Algorithm 1 Dictionary Learning Algorithm

Input: $\mathbf{Y}_i$, K, τ
Output:$\mathbf{D}$, $\mathbf{X}$
For time $t = 0$

Initialize $\mathbf{D}_0$: The initialization of $\mathbf{D}_0$ is done by selecting top K salient data points with

For time $t > 0$ until convergence (or a fixed number of iterations)

Sparse code update: While keeping the dictionary fixed, update x_i using (2.4).

Dictionary update: Keeping the sparse code fixed update each column of the dictionary by using (2.5) and solving

$$\min_{\mathbf{D}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2$$
$$\min_{\mathbf{d}_k} \|\mathbf{E}_k - \mathbf{d}_k \mathbf{X}^k\|_F^2 \quad \forall k$$

$\mathbf{d}_k$ is obtained by taking the singular value decomposition of $\mathbf{E}_k$=$\mathbf{U}\mathbf{\Sigma}\mathbf{V}$ and $\mathbf{d}_k = \mathbf{U}(:, 1)$. $\mathbf{X}^k$ is the k^{th} row of $\mathbf{X} \in \mathbb{R}^{K \times N}$

2.2 Applications of dictionary learning

As mentioned earlier, a number of applications have exploited the technique of dictionary learning. In image de-noising, in-painting, tracking, *etc.*, [6, 7] the basis functions are learned from image patches for local image analysis. Further, [9, 12, 13, 46–50] exploit the method for image classification purpose, where, the dictionary is learned to represent certain category of images using discriminative functions.

Initially dictionary learning algorithm was applied in image de-noising application, where the dictionary is learned from small patches extracted [6, 51] from a noisy image. The linear combination of the dictionary and the sparse codes were then exploited to provide a de-noised representation of the image. Methods have been developed for both gray scale [6, 51] and color images [32]. While de-noising has been one of the first applications to demonstrate the impact

of dictionary learning in obtaining sparse representation of the images, it has since been extensively used in other applications like image segmentation, image classification, visual tracking, event detection. In this section, we discuss these three applications of dictionary learning, which are more relevant to the works proposed in this thesis.

2.2.1 Local image analysis/segmentation

Local image analysis or image segmentation involves identifying objects in an image. Image segmentation aims at dividing an image in two distinct regions: foreground and background. Segmentation strategies using contour propagation have been studied and used for various image processing tasks over the last few decades. Parametric [25, 26, 52, 53] or geometric active contour [30, 54] models for segmentation are based on intrinsic image properties like intensity, edge, *etc.* Object boundary detection is performed either by explicit [25, 26, 53] or implicit [30, 55] motion of the curve such that the contour finally converges to the object boundary. Snake models [25, 26] are traditionally formulated as an optimization problem where an edge based external energy is generally used to drive the snake toward the object boundary. However, in many imaging scenarios obtaining quality edge maps is often difficult since an edge detector performance is susceptible to noise and clutter.

Mumford and Shah [56] proposed an alternative approach where the segmentation criteria relies solely on the gray values of the image pixels. This region based approach was made popular later by Chan and Vese [30], who used level sets to propagate geometric snakes to segment the image into sets of constant intensity regions. Due to the nature of this algorithm, Chan-Vese's method is also popularly known as a piecewise constant model. However, as noted in [57–59], the piecewise constant framework is inadequate to capture variation in region intensities, thus resulting in improper segmentation. While such local approaches can accommodate the inhomogeneities to an extent, choosing a proper window size for gathering region based statistics is often a non-trivial task. Therefore, based on the choice of the neighborhood size, such algorithms have a tendency to be either too local, or overly global. A

separate set of algorithms have been suggested [60, 61] which extends the piecewise constant idea in [30] to accommodate nonlinear intensity profiles. It may be debated that adding edge information to the region based framework can improve segmentation. However, extracting accurate edge map is a challenging issue by itself for such applications due the presence of speckle and clutter.

Segmentation of the imaged organs are carried out using a plethora of techniques like active contours, morphological operations and others [62–64]. In a recent paper, Mukherjee and Acton [54] introduced a method known as L2S, which generalizes the Chan-Vese model by approximating the foreground and background regions as piecewise polynomial functions, computed by linear combination of a few Legendre basis functions. Although the method proposed can handle intensity inhomogeneities to a greater extent, the segmentation quality relies heavily on the number of chosen basis functions.

Many other methods using dictionary learning based local image analysis and segmentation have been proposed in literature. In these methods, image segmentation is regarded as clustering of local regions based on some common properties or features and detect regions probable to belonging to an object or not. In [7, 65] the authors demonstrated a dictionary learning to cluster local image regions to further aid in image segmentation. The proposed approaches, model the data as a linear combination of low dimensional subspaces and the data that share same dictionary atoms are clustered together. Different clusters are allowed to share atoms in these methods making is possible to extend it to soft clustering. However, these methods require a initial cluster identification and final segmentation results are susceptible to these initialization methods.

It is also believed that segmentation quality can be improved by introducing prior information about the shape of the object [66–68]. This shape information is often introduced using statistical modeling [69, 70] of the object shapes in the training set. Various methods have exploited dictionary learning to obtain a prior information about the shape. Dictionary learned using pre-segmented lung images are exploited as shape priors for parametric active

contour models in [18, 71]. These methods have significant improvement in performance of segmentation, but the application of such algorithms are limited. Such techniques require pre-segmented objects to compute the shape model which is a tedious task, especially when one is dealing with a significant number of images. Moreover, since the methods learn the shapes of objects to be segmented, they perform as expected only for segmentation of pre-defined object types. It is worthwhile to mention that segmentation accuracy can be significantly enhanced by adopting supervised learning based methodologies [72, 73]. However, such algorithms rely extensively on manually annotated data.

2.2.2 Image classification

Classification in general is defined as a categorizing a new observation to any of the prior known categories. In computer vision classification is based on categorizing an image based on its contextual information. Classification can be supervised [74–76] or unsupervised [77–79] i.e., when prior information about the categories are present or absent respectively. In content based image classification techniques, in conjunction with designing a good classifier, extracting informative features are also necessary. The above mentioned methods generally extract feature descriptors from images using analytical formulation and using traditional classifiers such as support vector machines [80], neural networks [81] to extract decision boundaries for the dataset.

Recently, sparse representation based methods have been employed in supervised as well as unsupervised classification of images. As stated earlier, the sparse linear representation with respect to an over-complete basis projects the features in a higher dimensional space thus making the inherent image properties more distinctive. Additionally in supervised classification, the classifier model can be learned concurrently with the sparse codes.

Wright et al., [15] demonstrated in their seminal work on sparse coding based classification, a more compact representation can be obtained when an image is represented as a linear combination of similar images from the database. Such a data dependent basis function was

shown to handle outliers in a more efficient manner. The sparse representation technique for classification has been adopted for recognition of various objects including biometrics, digits, histo-pathological images [14, 48, 49].

Sparse representation based classification

Sparse representation based classification adopts the idea, that images of a particular object can be represented as a linear combination of similar images. In other words, each object category belong to a subspace. All data in this subspace can be represented as linear combination of the basis function that spans the subspace. Wright et al. [15] in their work showed the application of face recognition, where images of a particular subject obtained under various pose or lighting condition is treated as a category or a subspace. The theory presented in [15] states that a test sample can be compactly represented by choosing a few of the training samples. If $\mathbf{y_q}$ be the test image from class c and $\mathbf{D}_c = [\mathbf{y}_{c_1}, \ldots, \mathbf{y}_{c_n}]$ be the training samples of class c, then the test image will lie in the linear space spanned by $\mathbf{D}_c$. The test sample can then be written as a linear combination of all the training samples given by

$$\mathbf{y_q} = [\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_C]\mathbf{x_q}$$

$\mathbf{x_q}$ is the coefficients of linear combination and is non-zero at locations corresponding to atoms representing the category of $\mathbf{y_q}$. For classification, the sparse representation for the test image is first computed using the ℓ_1 relaxation, given by

$$\min_{\mathbf{x}_q} \|\mathbf{y}_q - [\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_C]\mathbf{x}_q\|_2^2 + \|\mathbf{x}_q\|_1 \tag{2.6}$$

Finally, the classification rule is determined based on minimum reconstruction error for a particular class

$$class(\mathbf{y}_q) = \min_c \|\mathbf{y}_q - \mathbf{D}_c(\mathbf{x}_q)_c\|_2^2 \tag{2.7}$$

$\|\mathbf{y}_q - \mathbf{D}_c(\mathbf{x}_q)_c\|_2^2$ is the residual error when represented with only a particular class features and $(\mathbf{x}_q)_c$ are the coefficients of $\mathbf{x}_q$ corresponding to the columns of $\mathbf{D}_c$. Sparse codes learned using fixed basis have also been exploited in shape based matching [35] using nearest neighbor classifiers. The sparse representation classification has been extended to other robust models, where the sparse codes are learned by incorporating other structural information about the data. Some works have integrated features descriptors via sparse coding [14, 82] in contrast to using only gray-scale intensity images. Others have introduced spatial constraints in the optimization problem which learn the sparse codes by exploiting spatial information between images in a dataset [83–86]

Sparse representation based classification has been proven to be very efficient in applications of image classification in handling illumination variations, pose variation and occlusion. However, there is always the issue of whether the images of an object category in a dataset are capable of representing all the images belonging to that class. To resolve this issue, learning the basis from training samples instead of using the images directly has proven to be more effective for classification purpose.

Discriminative dictionary learning

Dictionary learning is a technique for obtaining a data adaptive over-complete basis for a more compact representation. For image classification, a dictionary is learned for each class using the following in the training step

$$\min_{\mathbf{D},\mathbf{x}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 \text{ s.t. } \|\mathbf{x}_i\|_0 \leq \tau \ \forall \ i \tag{2.8}$$

For classifying a test sample, it uses the same strategy as shown in (2.6) and (2.7). In such a framework for dictionary learning in classification, a dictionary is required to learn for each class. For multi-class classification task, this becomes computationally expensive as the number of categories increases and additionally no discrimination is made between

dictionaries of different classes. Hence an attempt was made to learn the dictionaries for all the categories using a single cost function. However, to achieve this, some discrimination function between classes need to be incorporated in the optimization. The idea of including an inter class discrimination in the cost function was introduced in [9, 12, 13, 16, 87]. The general idea proposed in these papers usually involve solving an optimization of the following form,

$$\min_{\mathbf{D},\mathbf{x}} \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}, \mathbf{x}_i\|_2^2 + f(I_c, \mathbf{DX}) \text{ s.t. } \|\mathbf{x}_i\|_0 \leq \tau \ \forall i \tag{2.9}$$

Here, $f(I_c, \mathbf{D}, \mathbf{X})$ is inter class discriminatory function involving the dictionaries, sparse codes and class membership indicator function I_c. The main idea behind the discriminatory function is that features from same class share similar dictionary atoms which i.e., same category image representations will now belong to the same subspace. This also accomplishes that when represented with respect to the dictionary, the sparse representation are similar for data belonging to same class and significantly different between categories.

The linear model for dictionary learning has enabled in the design of more discriminative sparse codes by incorporating the intra-class discriminatory function. However, for image datasets, the inter-class separability is not necessarily linear. To accommodate the non-linearity in the data, some works have proposed a non-linear dictionary learning technique. The objective of these works is to obtain sparse codes and a dictionary from non-linearly transformed data [36, 88–91]. This achieves non-linear data separation in two levels: first in the kernel representation and then in the sparse coding step. The non-linearly transformed test image is then classified based on minimum residual error.

The dictionary learning techniques for classification mostly focus on generating discriminative representation. However, these do not incorporate any feature type selection or local region identification from images to boost the classification mechanism. Here in the thesis, we focus on selecting the more discriminative features from images and integrating it into the dictionary learning technique. The discriminative feature computation for image classification identifies more prominent features of an image or an image category. Simultaneously it takes

advantage of obtaining robust sparse representation using the dictionary learning algorithm.

2.2.3 Tracking and event detection from video

Object tracking and event detection problems are spatial-temporal sequence analysis problems. In video tracking [92, 93] the objective is to predict the motion of a moving object from a time sequence, while event detection [94–96] problems aim at detecting significant change occurring in a video. A number of issues need to be addressed in these problems e.g., pose variation, illumination changes, occlusion.

Object tracking in video

The traditional object tracking methods based on particle filter [92, 93, 97], kalman filter [98], impose a motion model on an object template. Based on the motion model and observation, the most likely position of the template is determined using probabilistic prediction models. These methods in general do not apply any variational model on the template which can handle illumination, pose variation or occlusion.

Similar to object recognition problems, sparse representation can aid in modeling the changes and variations in the object occurring due to affine transformations, illumination or occlusion. In sparse representation based visual tracking, the target is modeled as a linear combination of some dictionary or basis function which can capture the pose/illumination variation in addition to the motion models.

In [20, 34], the authors propose a model where the target to be tracked is represented as a linear combination of pre-detected target template and sample error set is used to model the occlusion. Although the paper shows promising results, depending on the dimension of the template, a large number of target templates are necessary to represent the new predictions appropriately. In [99], the authors extended the ℓ_1 norm to a mixed norm problem to accommodate interdependency between predicted templates. Additionally the dictionary is updated dynamically to adapt to the variations due to occlusion, illumination

and affine motion. In [19], the authors proposed a work for visual tracking under varying illumination. Here the intensity change of the target was represented as a sparse combination of Legendre polynomials, which are specifically designed for illumination change prediction on the template. These methods are designed to track or predict the present state of a target object based on past predictions and are not capable of detecting events or changes in scenes.

Abnormal event detection from video

Event detection from video is a broad field which can relate to any change in the temporal sequence that does not adhere to the original pattern of the sequence. This can indicate unusual or suspicious behavior of certain objects, malicious activity, change in usual trajectory patterns, sudden changes in scene due to accidents, *etc.* A sub problem in this category is abnormal, hazardous event detection which refer to unpredictable incidents or events that occur in a video abruptly and do not follow any regular pattern e.g., road accidents, sudden fire, *etc.*

Event or anomaly detection in videos is an active area of research in the field of image and video analysis. [100, 101] exploits motion analysis of object trajectories in detecting anomalies in motion pattern. In [102], the authors use a background subtraction framework for detecting fire. A clustering based method is used in [103], while [104–106] employs temporal analysis of spatial saliency maps in detecting anomalies in surveillance videos, crowded scenes. In [107], the authors analyze a graphical representation of a video to detect events and the extent of abnormality of the events.

Sparse and low rank representations have recently been used in event detection. In [33], the authors present a dynamic sparse coding technique to detect changes in scenes. In [108], the authors develop a weighted sparse representation scheme, while [109] uses dictionary learning on the histogram of maximal optical flow projection features of video frames. The ℓ_1 norm of the representation is used to predict abnormal frames.

Sparse coding with low rank representation has been a more popular model for event detection [110]. In such models, generally the event is designed as a sparse vector whereas the background is modeled as low matrix. The change in scene is obtained by taking absolute error between the observation and the estimated background. In [111], the authors exploit the low rank method in simultaneously detecting object motion and outliers. Whereas in [112] the authors analyze the trajectories in a low rank matrix completion and group sparsity framework to detect abnormalities in video. A class of structured sparsity is introduced in [113] to model the moving object while the background is modeled as a low rank matrix. Similar approaches were also proposed in [114, 115]. ADM [1] and DRMF [2] methods for anomaly detection also use a low rank and sparse representation based background subtraction method for detecting events in videos.

However, in scenarios where the videos are captured using hand-held or camera fitted with a car, camera jitter, motion of other objects lead to a dynamic background. In these scenarios, background is not static and thus background subtraction yields significant false positives. Thus background subtraction methods are not the preferred choice in such situations. A more sophisticated method which is able to handle significant background changes are desirable.

2.3 Feature Selection

Informative feature selection from a single image or for a particular category to obtain a more content specific descriptor of an image has been an interesting field of research. In order to obtain a image specific feature descriptor, one can aim at selecting from a pool of feature extraction methods (*feature type selection*). On the other hand, detecting local image regions which would describe the content of a single image (*relevant local feature selection*) is another approach to extract discriminative features from images.

2.3.1 Feature type selection

Depending on the complexity of the database items, it may be almost impossible to correctly represent every item based on a single feature. So far, to our knowledge, these exist no such analytic feature extraction technique which can extract significantly discriminative image characteristic universally across all object categories. This calls for feature boosting strategies, where multiple feature selection routines are combined to generate the feature set (or pool). An approach to solve for the intra-class scatter of image properties is to select the optimal set of features discriminative of a class. Such feature selection methods for enhancing image retrieval performance by retaining only the more informative features for a class via maximizing mutual information [116–118]. However, these methods use any one single feature type and hence suffer from a particular drawback which renders the above mentioned methods unreliable for classification, specifically for databases characterized by significant content variability.

In [119] a method of hierarchically arranging image features according to relevance for a particular class is discussed. The works in [120, 121] select subset of classifiers/predictors or a subset of optimal features for classification. While these methods have their specific advantages, the suffer from one common drawback. All these methods emphasize the selection of the optimal set of features using one particular analytical feature descriptor computation technique. These do not attempt to identify for an unknown image which is the optimal characteristics, color, texture or structure that differentiates it from the others.

2.3.2 Relevant local feature selection

While the feature type selection tries to identify the major characteristics present in an image, the low level local feature selection is often viewed as extracting the object of interest from an image and using the features from this region in further applications. A commonly practiced method in object detection is to perform segmentation to partition the image into foreground and background. A number of works [30, 52, 54] have been directed toward this

task. However, segmentation accuracy can suffer from variation in initialization. Additionally noise statistics, morphological complexities of objects and the presence of clutter often leads to a problem specific solution, which is difficult to generalize to a typical image retrieval or classification setting. This objective can be somewhat achieved by incorporating prior knowledge of the object of interest [17, 18]. Obtaining a robust shape model from the dataset requires a sufficient representation of all object shapes and morphologies, which is difficult to obtain specifically in applications of histo-pathological image analysis.

While segmentation routines tend to divide an image into foreground and background, salient object detection technique give weight to local image regions. The assigned weights are proportional to how probable a region is in belonging to an object of interest. Based on human visual attention model, salient object detection method [122–124] detect regions which are unique in a local neighborhood and capture human attention. Saliency is typically described as property of the image regions which makes them distinct from other regions and attracts human attention [124, 125]. Human visual attention has been an active field of research in psychology, neural sciences, biology. It has found various use in image analysis and computer vision because of its ability to improve performance in object detection, segmentation [126], recognition [127, 128], and tracking [129, 130]. Recent research on visual saliency detection can be broadly classified into two main approaches : top down and bottom up.

The **top down approach** exploits prior knowledge about object from a given dataset to obtain specific object from the image [131, 132]. Low-rank matrix recovery scheme in used in detecting salient regions in [133]. Here, an image is represented as a combination of low rank matrices that represent the background and noise while a sparse matrix represent the salient regions. However, these models rely on a prior knowledge of objects which is not readily available is all applications. These methods demonstrate acceptable results when a labeled dataset is available and the test images indeed contain objects from the training dataset.

The **bottom-up approach** on the other hand is based on key interest point or region detection depending on local contrast of various low-level image features. These low-level

image features are driven by intensity or color variation at each pixel, edges, gradients, spatial frequencies *etc.* or their combinations. The work proposed in [124] uses a combination of intensity values, spatial frequency, orientation in an image to detect local contrast based saliency in images. In [134] a bottom-up method using a coherent computational approach employing contrast sensitivity functions and center surround operation was presented. Others use global approaches - frequency domain [122] or graph [123, 135] analysis of the entire image to obtain the salient regions. The proposed algorithm in this paper performs saliency guided dictionary learning in an unsupervised scenario, hence the bottom up saliency detection approach is more suited for the methodology.

2.4 Similarity measure

Image classification or event detection, in addition to informative feature selection, designing a robust similarity measure has always been a critical step. Although sparse representation methods or Kernel methods makes the discriminatory pattern more prominent, the data may not be linearly separable. As a result the distance between two data points cannot be measured correctly by Euclidean distance. This calls for a more sophisticated and robust similarity measure design.

Similarity between images has been explored using global as well as local image features for applications such as image quality measurements, classification, retrieval, registration. In [136] the structural similarity of images has been used for image quality assessment. The authors in [137] employ a method of representing images as Gaussian mixtures and Kullback Leibler (K-L) divergence [138] between the estimated Gaussian mixtures for similarity computation in image retrieval. Various histogram matching approaches have been proposed to measure similarity between image features [139, 140]. An efficient earth mover's distance based similarity between histograms of image features was proposed in [141] for shape recognition and interest point matching between images.

Recently, some works have employed the compressibility of images in computing similarity. The compression based distance measures were shown to be parameter free unlike the above mentioned approaches. Additionally these methods take advantage of the sparsity in natural images. These approaches generally encode images with a particular compression technique e.g. *JPEG, JPEG2000, MPEG* and exploit the compressibility for computing similarity [142, 143]. The similarity distance are computed using formulations influenced from information theory [144, 145] for pattern matching. The authors in [146] use an encoder to convert media data to text and further use compression of the text data for retrieving images. In [147], the authors propose a compressor based on finite context models on intensity domain of images and uses normalized compression distance [145] to measure similarity.

More recent methods [10, 11] do not rely on available compression method. Instead, they use the information available form one image to encode other images and exploit the extent of compressibility in the cross representation to measure similarity. Since the dictionary learning method provides a compact representation of an image, the learned dictionary from patches of an image can be exploited as the optimal compressor of the image. The work of [10, 11, 148] uses this concept to design a similarity measure. In comparing a pair of images, one is represented with the dictionary of the other, and this forms the basis of the designed similarity measure in this thesis.

Chapter 3

Dictionary learning level set (DL2S)

The primary contribution of this work is to develop a region based segmentation framework which is capable of handling intensity inhomogeneity. The salient idea of this letter is to compute the optimal set of functions which can model the region intensities. In various imaging techniques such as ultrasound, fluorescence microscopy, limited power and processing ability degrade the image quality. Speckle is more prominent in such images, which are further degraded by contrast variation and heterogeneous illumination.

In a recent work, L2S [54], the authors developed a method which uses a set of pre-specified Legendre basis functions to perform region based segmentation of an object in presence of heterogeneous illumination. We hypothesize that in problems where a set of training images for the object is available for analysis (such as depth image sequence of blood vessels via ultrasound imaging), segmentation accuracy can be significantly improved by learning the basis functions instead of specifying them implicitly. Our solution to this problem involves the integration of a level set segmentation methodology with the dictionary learning framework.

Intensity inhomogeneity is a prominent problem in imaging applications like ultrasound or fluorescence microscopy [149]. In such scenarios, due to presence of noise, intra-object intensity inhomogeneity, accurate edge information is often not available since an edge detector's performance is susceptible to noise and clutter. Therefore, it is of considerable

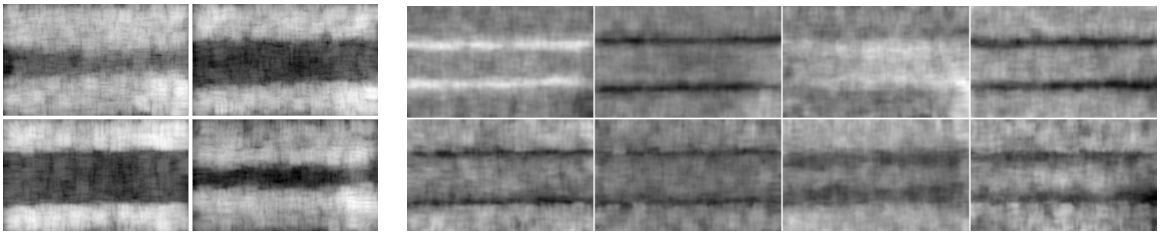
interest to develop a solely region dependent technique that can accommodate artifacts such as noise, clutter and illumination variation.

Some researchers have described local region based algorithms to tackle the intensity variations [57–59, 150]. In L2S [54], Legendre functions are used to compute the polynomial function that approximates the image region intensities. Despite its merits, L2S suffers from certain issues. First, the segmentation quality relies heavily on the number of chosen basis functions. Second, L2S suffers from scalability issues since the pre-specified bases cannot represent any arbitrary intensity variation. To cope with these drawbacks of regions based segmentation techniques, we propose a data-driven approach to model the intensity profile of the images which is capable of handling the intra-object intensity variations.

Objective

As it turns out, recent research in the field of sparse modeling and dictionary learning [6, 7, 15] have shown that for a given set of training data, one can obtain an optimal set of basis elements (atoms) to represent a signal. This is the main highlight of our proposed method– *instead of explicitly specifying the set of basis elements, we estimate an optimal set of bases from the set of training images using dictionary learning.* As mentioned earlier, segmentation accuracy can be enhanced by adopting supervised learning based methodologies [72, 73]. However, such algorithms rely extensively on manually annotated data. In contrast to such learning based methodologies, our solution does not require annotated data for segmentation.

To demonstrate our technique, we choose an important segmentation problem for ultrasound imaging. Blood vessels are imaged in C-mode using a portable, low cost, battery operated ultrasound device. Our objective is to segment the vessel boundary to assist medical practitioners for performing phlebotomy application such as intravenous needle placement. Images captured using these portable devices suffer from low contrast, noise and speckle in addition to inhomogeneous illumination of the objects which makes segmentation challenging.



(a) (b)

Figure 3.1: (a) shows sample images from a class and (b) shows dictionary learned from the images in that class

3.1 Methodology

The Chan-Vese framework [30] proposes to partition the image $f(\chi)$ ($\chi \in \Omega \subseteq \mathbb{R}^2$) into sets of constant illumination regions. The optimal partition is obtained by locally minimizing the following energy functional:

$$\int_{\Omega} |f(\chi) - c_1|^2 m_1(\chi) d\chi + \int_{\Omega} |f(\chi) - c_2|^2 m_2(\chi) d\chi \tag{3.1}$$

Here, we define ϕ as a level set function whose zero level set denotes the object boundary. ϕ is constructed such that its value is positive inside the zero level contour and negative outside. The local minimizer ϕ^* of (3.1) partitions the image such that the two region intensities are best approximated by the constant scalars c_1 and c_2, which are updated iteratively using alternating minimization. $m_1(\chi) = H_\epsilon(\phi)$ is the regularized version of the standard Heaviside function, the extent of regularization being controlled by the parameter ϵ, $m_2(\chi) = 1 - m_1(\chi)$.

3.1.1 Generalized Chan-Vese

As mentioned earlier, the constant intensity model fails to capture the intensity heterogeneity, common in most ultrasound imagery. To account for the nonlinear intensity profile, a

generalized version of Chan-Vese's model can be formulated as follows:

$$\hat{\mathcal{E}}(\phi, \mathbf{a}, \mathbf{b}) = \int_{\Omega} |f(\chi) - \sum_{i=0}^{k} a_i \mathbf{d}_i(\chi)|^2 m_1(\chi) d\chi + \int_{\Omega} |f(\chi) - \sum_{i=0}^{k} b_i \mathbf{d}_i(\chi)|^2 m_2(\chi) d\chi + \nu \int_{\Omega} |\nabla H_\epsilon(\phi)| d\chi + \lambda \left(||\mathbf{a}||_2^2 + ||\mathbf{b}||_2^2 \right) \tag{3.2}$$

Here $\mathbf{D}_k(\chi) = [\mathbf{d}_1(\chi), \ldots, \mathbf{d}_k(\chi)]^T$ is a dictionary which will be discussed in detail shortly. $\mathbf{d}_0(\chi) = \mathbf{1}$. $\mathbf{d}_1, \ldots, \mathbf{d}_k$ are dictionary elements or atoms which are used to model the non-linearity in the intra-region intensities of the images. The third term in (3.2) introduces smoothness in the solution, which is controlled using the parameter ν. $\mathbf{a} = [a_0, \ldots, a_k]^T, \mathbf{b} = [b_0, \ldots, b_k]^T$ are $(k+1)$ dimension real valued coefficient vectors. The parameter λ reduces over-fitting, by constraining the ℓ_2 norm of the coefficient vectors.

With $k = 0$, (3.2) reduces to the piecewise constant model in (3.1). In other words, (3.2) generalizes the traditional Chan-Vese technique by introducing capability to handle heterogeneous image regions. Here $\mathbf{d}_1, \ldots, \mathbf{d}_k$ can be interpreted as 'detail functions' to model the intensity variation in conjunction to the constant illumination term $\mathbf{d}_0$. As earlier, (3.2) can be optimized with respect to ϕ, $\mathbf{a}$ and $\mathbf{b}$ using alternating minimization.

Unlike [54], where the dictionary was pre-specified, we hypothesize that if a dataset of example images is available, we can enhance the segmentation performance by learning an optimal set of basis functions (dictionary elements) for region intensity approximation instead of using a pre-defined set of basis. For the application described in this paper, we are concerned with sets of ultrasound images, imaged using similar type of devices. The multi-depth images are captured at the same scale, and are preregistered. As a result, we have the provision to learn these functions $\mathbf{d}_i(\chi)$ directly from the dataset.

3.1.2 Dictionary Learning Level Set (DL2S)

Sparse coding techniques have gained popularity recently. Such algorithms have been used for a multitude of applications ranging from image denoising, inpainting, restoration, classification,

retrieval etc [6, 7, 15, 32]. Given a set of training data, the goal of dictionary learning is to compute a set of basis elements, also called *atoms*, such that each training data can be represented as a linear combination of only a few of these atoms. The key idea is to utilize the underlying sparsity of the training data, while minimizing the reconstruction error. Mathematically, if $\mathbb{F} = [f_1, \ldots, f_N]$ denotes the set of N discretized, vectorized and mean subtracted training images, we can use dictionary learning technique to compute the dictionary $\mathbf{D}_k = [\mathbf{d}_1, \ldots, \mathbf{d}_k]^T$ mentioned in (3.2) by solving the following optimization problem

$$\mathbf{D}_k = \arg\min_{\mathbf{D},\alpha_i} \sum_{i=1}^{N} ||f_i - \mathbf{D}^T\alpha_i||_2^2 \text{ s.t. } ||\alpha_i||_0 \leq \theta, \forall i \tag{3.3}$$

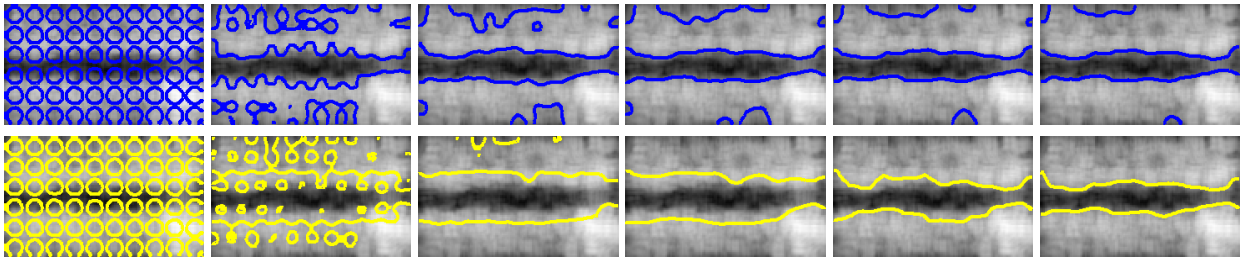
Here α_i is a coefficient vector corresponding to the i^{th} training image and θ is a scalar which dictates the level of sparsity. There are a number of methods in the literature that use some approximation to solve the hard optimization problem (3.3). For example, k-SVD [6] combines a greedy methodology using orthogonal matching pursuit algorithm to provide a fast solution to this problem. Dictionary learning exploits sparsity in the data (3.3) by constraining ℓ_0 norm of the coefficients.

3.1.3 DL2S curve evolution

Let us denote $\hat{\mathbf{D}}_k = [\mathbf{d}_0(\chi)^T\ \mathbf{D}_k(\chi)]^T$. We first try to minimize (3.2) with respect to $\mathbf{a}$ and $\mathbf{b}$, by taking derivatives and setting the result to zero. A closed form solution is obtained as follows:

$$\hat{\mathbf{a}} = [\mathbf{K} + \lambda\mathbb{I}]^{-1} \int_\Omega \hat{\mathbf{D}}(\chi) f(\chi) m_1(\chi) d\chi \tag{3.4}$$

$$\hat{\mathbf{b}} = [\mathbf{L} + \lambda\mathbb{I}]^{-1} \int_\Omega \hat{\mathbf{D}}(\chi) f(\chi) m_2(\chi) d\chi \tag{3.5}$$



(a) (b) $t = 10$ (c) $t = 20$ (d) $t = 40$ (e)$t = 70$ (f)$t = 140$

Figure 3.2: (a) shows initialization of the L2S (top row, blue) and DL2S (bottom row, yellow) curve. (b), (c), (d), (e) and (f) show the curve evolution at $t = 10$, 20, 40, 70 and 140 respectively

where [.] denotes a matrix. $\mathbf{K}$ and $\mathbf{L}$ are $k \times k$ Gramian matrices [151], in which the $(i,j)^{\text{th}}$ entries are obtained as

$$[\mathbf{K}]_{i,j} = m_1(\chi)\,\langle \mathbf{d}_i, \mathbf{d}_j \rangle \text{ and } [\mathbf{L}]_{i,j} = m_2(\chi)\,\langle \mathbf{d}_i, \mathbf{d}_j \rangle \tag{3.6}$$

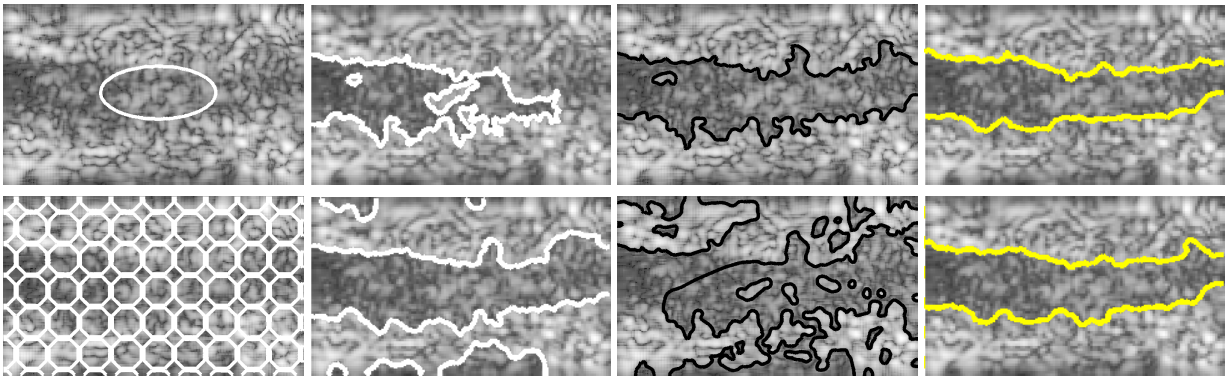
$0 \leq i,j \leq k$ and $\langle , \rangle$ denotes the Euclidean inner product operator. With the updated coefficient vectors, we can now minimize (3.2) with respect to ϕ using variational calculus. We obtain the following partial differential equation using gradient descent technique for minimization.

$$\frac{\partial \phi}{\partial t} = \left[-|f(\chi) - \hat{\mathbf{a}}^T \hat{\mathbf{D}}_k(\chi)|^2 + |f(\chi) - \hat{\mathbf{b}}^T \hat{\mathbf{D}}_k(\chi)|^2\right] \delta_\epsilon(\phi) + \nu \delta_\epsilon(\phi) \text{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right) \tag{3.7}$$

Where $\delta_\epsilon(\phi)$ is a regularized version of the Dirac delta function. We initialize $\phi|_{t=0} = \phi_0$ and $\frac{\delta_\epsilon(\phi)}{|\nabla \phi|} \frac{\partial \phi}{\partial \hat{n}} = 0$ at the domain boundary. The gradient flow of DL2S is computed iteratively by discretization of (3.7) using a finite difference scheme.

3.2 Experimental Results

We use five different sets of images to evaluate the performance of our algorithm. Out of them, three datasets contain images of medical phantoms which mimic human veins. These phantoms are generally used by medical practitioners for device calibration. The



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

Figure 3.3: Comparison of segmentation results using manual and automatic initialization methods. (a) initialized contour (b) segmentation results of Chan-Vese (white), (c) segmentation via L2S (black) and (d) segmentation via DL2S model (yellow)

remaining two datasets consists of human vein images, captured *in vivo*. Each dataset contains approximately 18 to 60 images, captured in C-mode using a portable, battery operated ultrasound scanner. The different images in a given set correspond to the image of a vein at various depths. Note that each dataset consists of registered blood vessel images. The vessel orientation and scale are also consistent. A separate dictionary is computed using the mean subtracted images for each of the datasets.

Dependency on contour initialization: We show the performance of our algorithm using both manual and automatic initialization methods. The segmentation results with manual and automatic initialization for Chan-Vese [30], L2S [54] and DL2S are shown in Fig. 3.3 for the same image. We observe that the segmentation performance of L2S drops significantly for automatic initialization, which is also true for Chan-Vese method. In comparison DL2S has similar segmentation results for both initialization technique. In Fig. 3.2, the evolution steps for L2S [54] and DL2S is shown. As noticed, from the experiments on ultrasound image datasets, DL2S converges faster than L2S.

Dependency on dictionary size: We perform sensitivity analysis experiment to study the performance of the segmentation algorithm with changing dictionary size. The Dice indices are plotted (along Y-axis) for L2S [54] (Fig. 3.5 (a)) ans DL2S (Fig. 3.5 (b)) to show

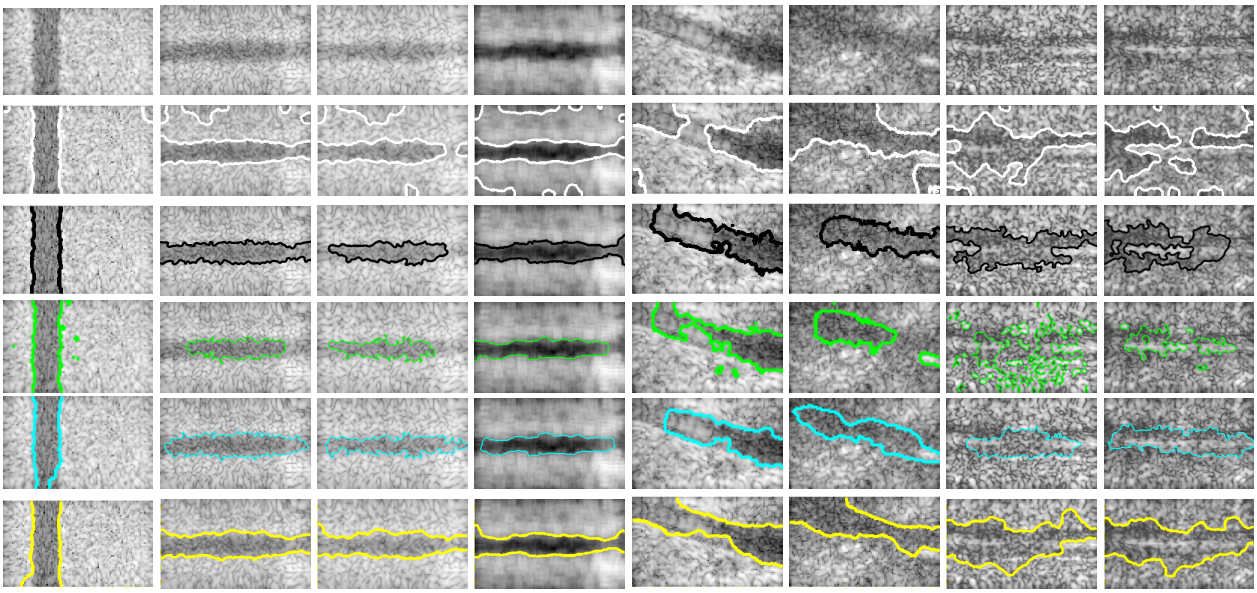


Figure 3.4: Segmentation comparison of DL2S other region based methods is shown. The original C-mode ultrasound images captured with a portable scanner are shown in the first row. Rows 2, 3, 4 and 5 show the segmentation results using the methods of Chan-Vese [30] (white), L2S [54] (black), Li *et al.* [57] (green) and Lankton *et al.* [58] (cyan). The last row shows the performance of DL2S (yellow).

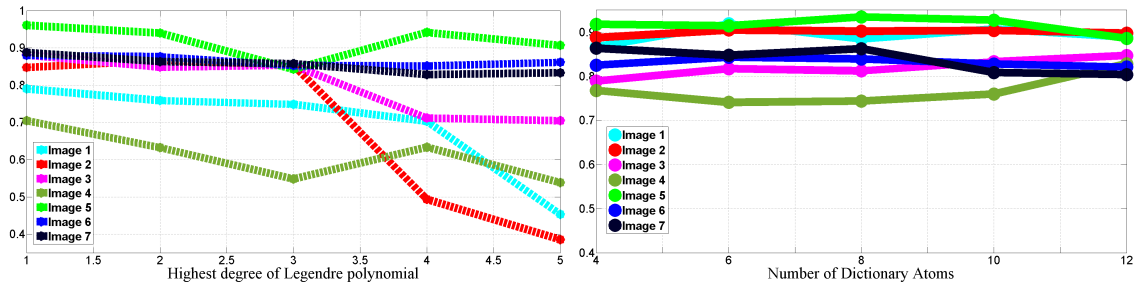
the performance with changing basis/dictionary size (along X-axis) for 7 randomly chosen images. In comparison to L2S, where performance decreases with increasing number of basis functions, DL2S exhibits a more stable performance. Based on experiment evaluation, we fix the number of dictionary elements $k = 8$ which is at most 50% of the size of the smallest dataset. We choose sparsity inducing parameter $\theta = 3$ such that about 30% or less number of atoms can be used for representing the training images.

Quantitative comparison of segmentation: Fig. 3.4 shows the segmentation performance

Table 3.1: Quantitative Comparison

Dataset	*DL2S*	*Chan-Vese* [30]	*L2S* [54]	*Lankton* [58]	*Li* [57]
(i)	**0.93±0.02**	0.91±0.07	0.89±0.09	0.83±0.06	0.86±0.08
(ii)	**0.90±0.04**	0.88± 0.05	**0.90±0.06**	0.70±0.09	0.71±0.12
(iii)	**0.86±0.08**	0.85±0.11	0.85±0.12	0.65±0.12	0.56±0.12
(iv)	**0.83±0.06**	0.73±0.12	0.70±0.19	0.72±0.05	0.67±0.04
(v)	**0.76±0.10**	0.75± 0.14	0.72±0.16	0.73±0.12	0.72±0.15

of the methods due to Chan-Vese (white) [30]), L2S [54] (black), Li *et al.* (green), Lankton



(a) (b)

Figure 3.5: (a) Dice index for L2S with changing number of basis functions, (b) Dice index for DL2S with changing size of dictionary.

et al. (cyan) and, DL2S (yellow) in rows 2-6 of Fig. 3.4, respectively. Qualitative results suggest that DL2S is able to capture the blood vessels more appropriately in presence of severe contrast and intensity inhomogeneity. A quantitative comparison for five datasets ((i) − (v)) are shown in Table 3.1. The Dice index, $D = \frac{2|s_t \cap s_g|}{|s_t| + |s_g|}$, is evaluated for all the aforementioned algorithms. Here s_g denotes the ground truth segmentation (achieved by human experts) and s_t is the solution using an automated algorithm. The average Dice indices, along with the standard deviations are shown in Table I. Each row in the table corresponds to the Dice value for a particular dataset, for each algorithm.

On average, using DL2S, we observe an increase in segmentation accuracy by more than 12% over all the sets of ultrasound images. Additionally, it should be noted here, for a particular dataset, the variability in the classification accuracy is significantly small in comparison to the competing methods. We also evaluated the computational time, whereby DL2S was slightly more expensive than [30], [54] and slightly less expensive than [57] and [58]. On a Windows-7 PC with 16GB RAM and Intel i7 processor. On a 240×400 image, the average computational time (sec) for DL2S is 15.34 as compared to 5.27, 9.54, 36.4 and 19.13 for L2S, Chan-Vese, Lankton *et al.* and Li *et al.* respectively.

3.3 Discussion

The Chan-Vese method performs segmentation by approximating an image $f(\chi)$ by a piecewise constant image $g(\chi)$. To make the model more flexible, we add higher order terms which can capture the intensity variations in the regions. Going by the intuition of Chan and Vese, it is fair to approximate the mean image of a dataset as a piecewise constant image.

Assuming a mean image which is approximately piecewise constant, the dictionary atoms learned from the mean subtracted dataset can be utilized to provide the non-linear variation necessary to model the intensity inhomogeneity. The energy functional in (3.2) essentially incorporates this idea in a mathematical framework. One can also think of the dictionary atoms as incorporating higher order details, learned to suit the dataset. The dictionary atoms aid in retaining the more significant image properties and compactly represent the dataset.

DL2S is applicable where a set of pre-registered training data is available, for example multi-depth ultrasound images of blood vessels, in temporal image sequences of biomedical objects such as carotid artery, heart videos. In applications involving a temporal image sequence, the first few frames of the video can be treated as the training data to learn the dictionary.

Chapter 4

Image classification via feature nomination

Standard image retrieval or classification techniques generally follow a two-step approach. First, in the training step, a set of discriminative features are chosen to represent an image which are exploited to learn a classifier. In the second step: the validation or test step, the same set of discriminative feature descriptors is chosen to represent the test image, and the features are then input to the classifier model which determines the category of the test image. As described earlier, identifying the features which are more relevant in distinguishing between different category images is a crucial task in the application of classification. The performance of the classifier models rely highly on the image features, which generally emphasize the color, texture or frequency content of the image. An appropriate choice of the features can boost the performance of the classifier significantly. Due to the intra-class and inter-class content variability in natural image classification, a single feature type is not sufficient to capture all the informations correctly. This calls in for a *meta-algorithm* [12] which would choose automatically the more relevant feature descriptor for an image from a given collection of feature types.

In the following sections we discuss a method for feature nomination by adopting a

dictionary learning based classification technique. The main objective of this work is to design a mutual information based score for feature nomination for a particular test image.

For datasets demonstrating considerable variability in contents of the images of same or different category, the task of selecting one representative feature type is often non-trivial. Depending on the complexity of the database items, it may be almost impossible to correctly represent an item based on a single feature type. This is chiefly because one particular type of feature descriptor may not be sufficiently discriminative for all the categories of objects present in the database. This calls for feature boosting strategies, where multiple feature selection routines are combined to generate the feature vector set.

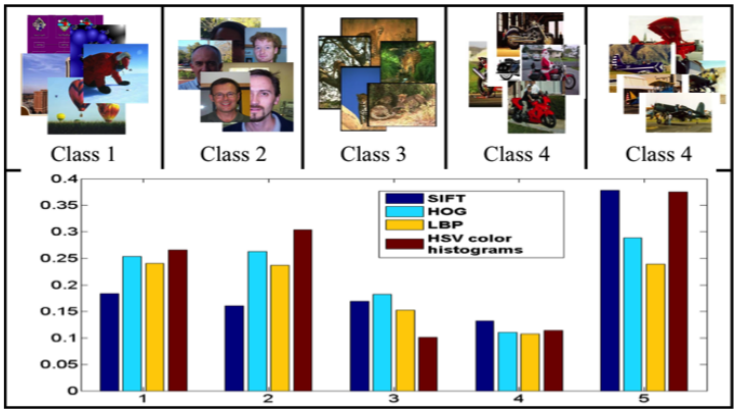


Figure 4.1: Overview of supervised image classification

Some methods [116–118] employ a method to retain more informative local features by maximizing mutual information for enhancing image retrieval. However these methods use any one single feature type and hence suffer from a particular drawback which renders the above mentioned methods unreliable for classification and retrieval purposes, especially for databases characterized by significant content variability.

As shown in Fig. 4.1, for different classes, classification accuracy changes with the feature type. With greater intra-class complexity, feature descriptors extracted by one particular method may not be discriminative enough to represent one class. Motivated by this fact, we design a system, which is capable of choosing the appropriate feature given a test image for accurate classification based on sparse representation.

Objective

In this chapter we aim at developing a method for designing compact and class-specific dictionary that can be utilized for classification. The original features can then be represented as a linear combination of this dictionary where the features from the same class share a common dictionary atom making it more class distinctive. Simultaneously, from this dictionary learning algorithm, we obtain a classifier weight matrix. This is used to interpret the sparse codes of the test image and assign a class label. A relevance measure between features and the class to which they belong can be obtained by maximizing mutual information between the test feature and the class features. So, finally for a given test image, once the sparse codes for different feature types and corresponding class labels are determined, we deploy an information theoretic technique for selecting the most relevant feature. The final classification is obtained using the nominated feature descriptor. An overview of the image classification system for meta-algorithm is shown in Fig. 4.2.

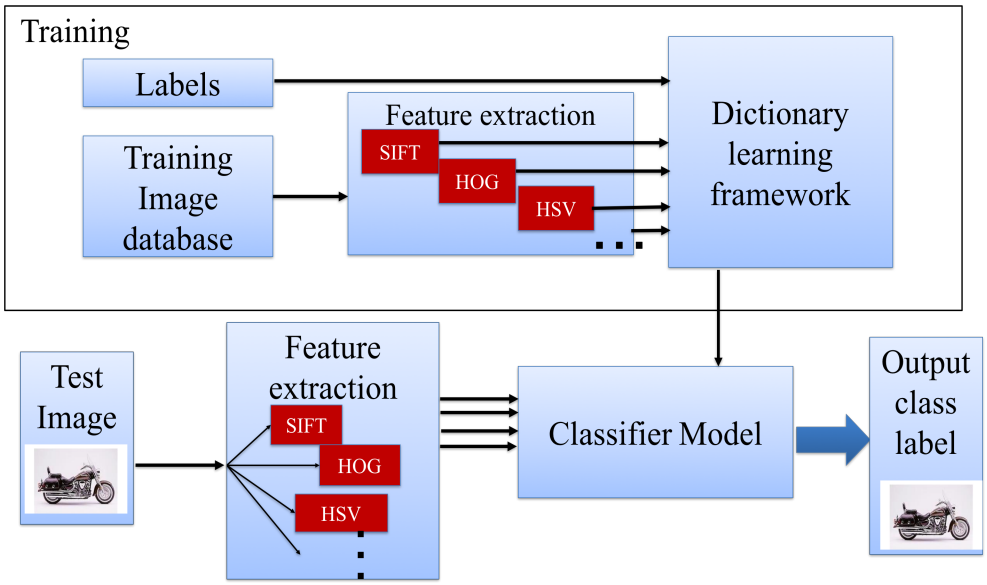


Figure 4.2: Overview of supervised image classification

4.1 Meta-algorithm for feature nomination

For designing the meta-algorithm for feature selection, we employ the discriminative dictionary learning based classification scheme.

Notations: Let us define a matrix $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_C]$, where C is the number of classes present in the dataset. Here $\mathbf{Y}_i = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_i}]$, $(\mathbf{Y}_i \in \mathbb{R}^{n \times N_i})$. $\mathbf{y}_v \in \mathbb{R}^n \times 1$ denotes a feature vector for v^{th} image in i^{th} class containing N_i images, i.e., $v = 1, 2, \ldots, N_i$. The dictionary $\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_C]$ is learned from the set of training examples and $\mathbf{D}_i \in \mathbb{R}^{n \times K}$ is the sub-dictionary representative of each class. Let $\mathbf{x}_v \in \mathbb{R}^M$ $(M = KC)$ be the sparse code for representing $\mathbf{y}_v$. The sparse codes for a class can be embedded in the matrix $\mathbf{X}_i = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_i}] \in \mathbb{R}^{M \times N_i}$. $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_C]$, denote the sparse codes for the dataset.

4.1.1 Discriminative dictionary learning and classification

As shown in Section 2.2.2, the discriminative dictionary learning aims to learn a single dictionary instead of separate dictionaries for each class. In doing so, a discrimination function need to be introduced in the learning framework to make the sparse codes discriminative. The purpose is to build class representative dictionary, so that sparse codes generated for features belonging to the same class, share similar dictionary atoms [9, 13]. The following optimization to obtain the desired dictionary.

$$\min_{\mathbf{X},\mathbf{D},\tilde{\mathbf{A}},\mathbf{W}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \alpha\|\mathbf{Q} - \tilde{\mathbf{A}}\mathbf{X}\|_F^2 + \beta\|\mathbf{H} - \mathbf{W}\mathbf{X}\|_F^2 \text{ s.t. } \|\mathbf{x}_v\|_0 \leq t \ \ \forall\, v \tag{4.1}$$

$\mathbf{Q} = [\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_C]$, $\mathbf{Q}_i \in \mathbb{R}^{KC \times N_i}$, is the label determining the pair of dictionary atom and signal sharing the same class. $\mathbf{Q}_i(a, b) = 1$ if $\mathbf{d}_a$ is a dictionary atom used for representing class i and $\mathbf{y}_b$ is a training data from the same class i. $\tilde{\mathbf{A}}$ is a transformation matrix that would regularize the sparse codes of the same class to share similar dictionary atoms. $\mathbf{H}$ is the matrix containing the class labels i.e., $\mathbf{H}(i, b) = 1$ if $\mathbf{y}_b$ is a member of class i. Assuming

a linear classifier model; the label of an input signal is given as:

$$(l(\mathbf{y}_v) = i) = \max_i \mathbf{W}\mathbf{x}_v \tag{4.2}$$

$\mathbf{W}$ is the classifier determinant parameter, which regularizes the sparse codes from same class to share similar dictionary atoms.

Optimization The dictionary is first initialized by selecting random columns for the data $\mathbf{Y}$. Then the dictionary and the sparse codes are updated by solving the dictionary learning step without the class discrimination constraints given as follows,

$$\min_{\mathbf{X},\mathbf{D}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \text{ s.t. } \|\mathbf{x}_v\|_0 \leq t, \ \forall \ v \tag{4.3}$$

The optimization is solved using the K-SVD algorithm [6] as discussed in (2.5) in Section 2.1. This $\mathbf{D}$ and $\mathbf{X}$ serves as the initial dictionary and sparse codes in the discriminative dictionary learning method. $\tilde{\mathbf{A}}$ and $\mathbf{W}$ are initialized by solving the following equations,

$$\begin{aligned} \min_{\tilde{\mathbf{A}}} \quad & \|\mathbf{Q} - \tilde{\mathbf{A}}\mathbf{X}\|_2^2 + \lambda_1 \|\tilde{\mathbf{A}}\|_2^2 \\ \min_{\mathbf{W}} \quad & \|\mathbf{H} - \mathbf{W}\mathbf{X}\|_2^2 + \lambda_1 \|\mathbf{W}\|_2^2 \end{aligned} \tag{4.4}$$

A closed form solution of both $\tilde{\mathbf{A}}$ and $\mathbf{W}$ can be obtained by solving the above equations given as follows,

$$\begin{aligned} \tilde{\mathbf{A}} &= \mathbf{Q}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda_1 \mathbb{I})^{-1} \\ \mathbf{W} &= \mathbf{H}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \lambda_2 \mathbb{I})^{-1} \end{aligned} \tag{4.5}$$

$\mathbb{I}$ is an identity matrix $\in \mathbb{R}^{M\times M}$. The three functions of $\mathbf{X}$ in the optimization can be combined and the optimization equation can be written as follows

$$\min_{\mathbf{X},\mathbf{D},\tilde{\mathbf{A}},\mathbf{W}} \left\| \begin{array}{ccc} \mathbf{Y} & - & \mathbf{DX} \\ \sqrt{\alpha}\mathbf{Q} & - & \sqrt{\alpha}\tilde{\mathbf{A}}\mathbf{X} \\ \sqrt{\beta}\mathbf{H} & - & \sqrt{\beta}\mathbf{WX} \end{array} \right\|_F^2 \text{ s.t. } \|\mathbf{x}_v\|_0 \leq t,\ \forall\, v \tag{4.6}$$

This can be written as

$$\min_{\mathbf{X},\hat{\mathbf{D}}} \|\hat{\mathbf{Y}} - \hat{\mathbf{D}}\mathbf{X}\|_F^2 \text{ s.t. } \|\mathbf{x}_v\|_0 \leq t,\ \forall\, v;$$
$$\text{where } \hat{\mathbf{Y}} = \begin{bmatrix} \mathbf{Y} \\ \sqrt{\alpha}\mathbf{Q} \\ \sqrt{\beta}\mathbf{H} \end{bmatrix} \text{ and } \hat{\mathbf{D}} = \begin{bmatrix} \mathbf{DX} \\ \sqrt{\alpha}\tilde{\mathbf{A}}\mathbf{X} \\ \sqrt{\beta}\mathbf{WX} \end{bmatrix} \tag{4.7}$$

the (4.7) can be solved by using K-SVD algorithm [6] as shown in algorithm:1. The columns of $\mathbf{D}$ are normalized after each iteration. $\mathbf{D}$, $\mathbf{W}$ and $\tilde{\mathbf{A}}$ are normalized using the euclidean norm of the columns of $\mathbf{D}$ i.e., $\mathbf{D} = [\frac{\mathbf{d}_1}{\|\mathbf{d}_1\|_2^2}, \ldots, \frac{\mathbf{d}_K}{\|\mathbf{d}_K\|_2^2}]$, $\mathbf{W} = [\frac{\mathbf{w}_1}{\|\mathbf{d}_1\|_2^2}, \ldots, \frac{\mathbf{w}_K}{\|bld_K\|_2^2}]$ and $\tilde{\mathbf{A}} = [\frac{\mathbf{a}_1}{\|\mathbf{d}_1\|_2^2}, \ldots, \frac{\mathbf{a}_K}{\|\mathbf{d}_K\|_2^2}]$.

4.1.2 Mutual information based feature nomination

Our goal is to devise a method or automatically choose the more relevant feature for a query image. To achieve that, we propose an information theoretic approach to dynamically choose the feature descriptor based on a given query type and the image contents. A relevance measure between features and the class they belong to can be obtained by maximizing the mutual information [116, 117]. For a given feature x the mutual information between the feature and its class, $l(x) = i$, is given by.

$$\mathcal{I}(\mathbf{x}, l(\mathbf{x}) = i) = \mathscr{H}(i) - \mathscr{H}(i|\mathbf{x}) \tag{4.8}$$

where $\mathscr{H}(\mathbf{x})$ is the entropy of $\mathbf{x}$ given by,

$$\mathscr{H}(\mathbf{x}) = p(\mathbf{x}) \log(\frac{1}{p(\mathbf{x})}) \tag{4.9}$$

For any class i the class probability is given as, $p(i) = \frac{N_i}{N}, i = 1, 2, \ldots, C$. If number of training images per class constant, that implies the entropy of a class is also constant. Thus maximizing the mutual information between a feature and a class would mean minimizing the conditional entropy $\mathscr{H}(i|\mathbf{x})$, which is given as:

$$\mathscr{H}(i|\mathbf{x}) = p(i|\mathbf{x}) \log \frac{1}{p(i|\mathbf{x})} = \frac{p(\mathbf{x}|i)p(i)}{p(\mathbf{x})} \log \frac{p(\mathbf{x})}{p(\mathbf{x}|i)p(i)} \tag{4.10}$$

The class conditional probability measure for a feature can be estimated by using a Parzen window technique using a Gaussian kernel. Thus $p(\mathbf{x}|i) = \frac{1}{N_i} \sum_{v=1}^{N_i} \mathscr{K}(\mathbf{x} - \mathbf{x}_v, \Sigma)$. Where, $\mathscr{K}(\mathbf{x} - \mathbf{x}_v, \Sigma) = \frac{1}{2\pi^{\frac{M}{2}} |\Sigma|^{\frac{1}{2}}} e^{-(\mathbf{x}-\mathbf{x}_v)^T \Sigma^{-1} (\mathbf{x}-\mathbf{x}_v)}$. $\mathbf{x}_v$ refers to a member of the training data of class i and the marginal is given as $p(\mathbf{x}) = \Sigma_{i=1}^{C} p(\mathbf{x}|i)p(i)$. When a feature descriptor for the test data $\mathbf{x}$ and its class label i is available, the mutual information provides a measure of certainty of $\mathbf{x}$ belonging to class i.

4.1.3 Image classification by meta-algorithm

We define a feature descriptor type F_l where $l = 1, 2, \ldots, L$ and L denotes the number of feature types being used for classification. For our experiments we use four features F_1: SIFT [22], F_2: Histogram of oriented gradients (HOG) [23], F_3: local binary pattern (LBP), and F_4: HSV color histograms. We use our feature nomination algorithm to choose between these four features to provide the ultimate classification result. The feature vector $\mathbf{Y}^l = [\mathbf{Y}_1^l, \mathbf{Y}_2^l, \ldots, \mathbf{Y}_C^l]$ corresponds to feature type l, for classes $1, 2, \ldots, C$. The respective sparse codes are $\mathbf{X}^l = [\mathbf{X}_1^l, \mathbf{X}_2^l, \ldots, \mathbf{X}_C^l]$. The sparse codes for a particular feature descriptor

l is obtained by solving the following

$$\min_{\mathbf{X}^l,\mathbf{D}^l,\tilde{\mathbf{A}}^l,\mathbf{W}^l} \|\mathbf{Y}^l - \mathbf{D}^l\mathbf{X}^l\|_2^2 + \alpha\|\mathbf{Q} - \tilde{\mathbf{A}}^l\mathbf{X}^l\|_2^2 + \beta\|\mathbf{H} - \mathbf{W}^l X^l\|_2^2 \tag{4.11}$$

As the number of features in the training set remains the same irrespective of the feature descriptor type, $\mathbf{Q}$, $\mathbf{H}$ which correlate between the features and their classes, remain same. For a given query image q, the feature descriptor $\mathbf{y}_q^l$ for feature type l is computed and the respective sparse code $\mathbf{x}_q^l$ is obtained by solving, $\min_{\mathbf{x}_q^l} \|\mathbf{y}_q^l - \mathbf{D}^l\mathbf{x}_q^l\|_2^2$ s.t $\|\mathbf{x}_q^l\|_0 \leq t$ The feature specific class label for the test image is given by $(l(\mathbf{x}_q^l) = i) = \max_i ((\mathbf{W}^l)\mathbf{x}_q^l)$.

Once the class labels corresponding to the feature descriptors F_l are obtained, it is required to identify the most relevant class for the query. Comparing the class conditional densities, a measure of how likely the test image will actually belong to the class label assigned to it, can be obtained. The class conditional entropy is computed for the sparse codes $\mathbf{x}_q^l$, and compare we compare $\mathscr{H}(\mathbf{x}_q^l|l(\mathbf{x}_q^l))$ for all l. Thus the final classification result is given by the nominated feature type l :

$$l(q) = \min_l \mathscr{H}(\mathbf{x}_q^l|l(\mathbf{x}_q^l)) \tag{4.12}$$

Image features

1. **SIFT**: *Scale invariant feature transform* [22] is a feature extraction method by identifying key-points from images and extracting features from a local region around the key-point. These key-points are detected by analyzing the image in scale space. The image is first convolved with Gaussian filter with different standard deviation (scale). The difference of a Gaussian is obtained from differencing adjacent Gaussian blurred images. The key point is then identified as the maximum in a local region and along the scale. Histograms of the gradient magnitudes over a local region around the key-point is computed. The descriptors obtained are rotation, illumination and affine transformation

invariant. SIFT has been used in a number of applications like object recognition, key-point matching, finding correspondence in images etc.

2. **HOG**: *Histogram of oriented gradients* [23] computes the image gradient in local regions or blocks. The gradient magnitudes are accumulated in a histogram with bins ranging from 0-180 or 0-360 degrees. Creating the histograms in this manner contain the information about the gradient magnitudes and their corresponding orientation. HOG has been found to be particularly useful in detection of detecting humans, vehicles, animals etc. which demonstrate distinctive structure.

3. **LBP**: *Local binary pattern* [27] is a method for extracting local texture information from images. In a local block, the neighborhood of each pixel is analyzed in clockwise of anti-clockwise direction and a binary number of 0 or 1 is assigned to the pixels if it is less or greater than the center pixel respectively. The sequence of 0s and 1s gives a binary number. These binary numbers associated with each pixel in the local regions are used to create a histogram. LBP has been shown to be useful in texture classification.

4. **Color Histogram**: The color histogram are created by looking at the intensity profile of the image and creating a histogram using the frequency of occurrence of each of the intensity values. To generate a histogram using all the three color channels, the RGB values for each pixel need to be considered.

Dataset and results

Experiments were performed using the Caltech 101 dataset, which contains (Fei-Fei, Fergus and Perona) 101 different categories with 9,144 images. Sample images from the dataset are shown in 4.3. The number of images in a class varies from 31 to 800. We choose randomly selected 28 images per class to train the classifier for each of SIFT, HOG, LBP and HSV color histograms. The remaining images were used as test images.

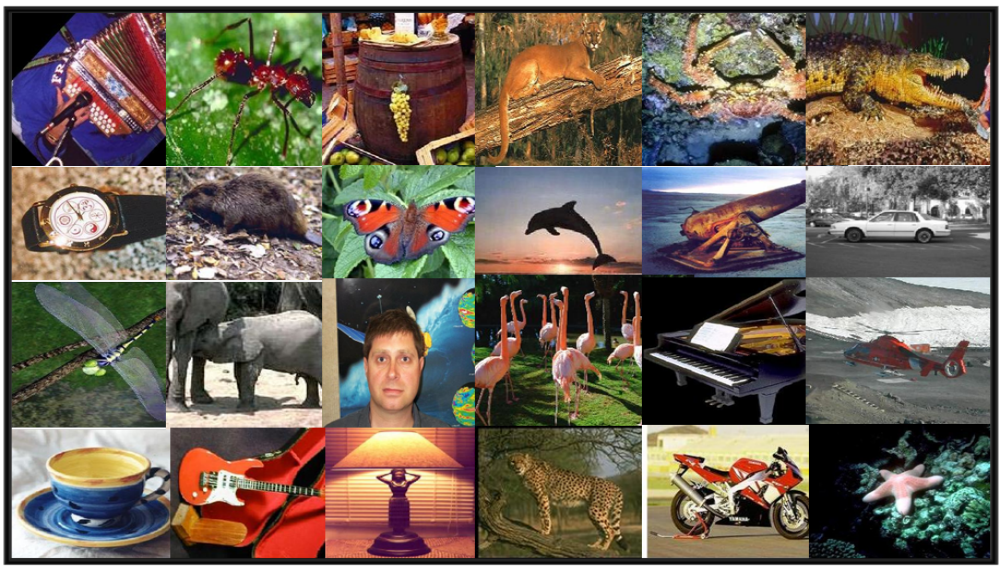


Figure 4.3: Sample images for the Caltech 101 dataset

For SIFT we extract the features in similar lines with [13]. We first compute the SIFT features on 16x16 grid with spacing of 2 pixels. Then we compute the spatial pyramid [139] structure for 3 levels, breaking the image into 4 blocks and then into 8 blocks. Then, the dimensionality of the extracted features was finally reduced using PCA. For HOG features, we compute the spatial pyramid by concatenating the histograms of the first, second and third level i.e., by breaking the image in 1x1, 3x3 and 5x5 blocks. Similar features were computed using LBP and color histograms, but only two levels were used to create the spatial pyramid structure. The sparse codes and the class labels we obtained using these four features. Finally the feature descriptor voting using the conditional entropy was accomplished using these sparse codes and the features for the obtained class labels.

In Fig. 4.4(a), we show accuracy percentage using feature descriptor voting scheme for classes which have accuracy more that 50%. The accuracy % is calculated as $\frac{\#\,images accurately\,classified}{\#\,images\,in\,the\,class}$. About 10% of the classes for the dataset have 100% accuracy and 12.7% classes have more than 90% accuracy. Assuming that accurate class labels will be obtained for at the least one of the feature descriptor type, out feature voting scheme chooses the correct class for 88.93% cases. A comparison using the bagging predictor [120] with our classification

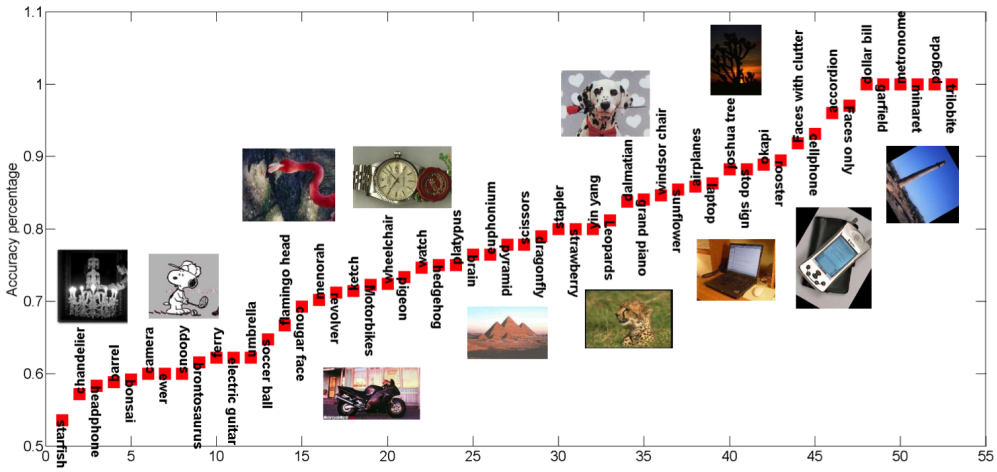


Figure 4.4: The figure shows classification accuracy for classes with greater that 50% accuracy

algorithm is shown in Figure 4. In our case, once the class label for each feature is obtained using the predictor, the optimal class is chosen when at least two of the sub-classifiers have identified the same class. Our method consistently gives a better result with an average 20% improvement in accuracy.

4.1.4 Discussion

Here we show a discriminative dictionary learning based classification scheme and introduced a information theoretic feature nomination algorithm to automatically decide the more discriminative feature for the query image. Our method described here chooses the most distinctive query specific feature for more accurate classification and at the same time does not require comparing the query feature with all the training features.

However, it employs a linear classifier model and is not able to capture the non linearity in the data. This problem can be addressed by non-linearly transforming the data (see appendix A) and learning the dictionary from the nonlinearly transformed data. The method discussed in (see appendix A) modifies the features nomination to a feature combination paradigm where the contribution of each feature type is determined by a information theoretic measure.

The **Meta-algorithm** extracts the features from the images by prioritizing all the regions in the image uniformly i.e., does not account for the object of interest in the image. In addition

it requires pre-annotated images or categories for learning the classifier. In applications where large datasets of annotated images are not available, the classifier often leads to over-fitting. In such scenarios local discriminative features for each image needs to be identified. To deal with this type of scenarios, a saliency guided dictionary learning technique that prioritize local features to boost the classification system is developed and discussed in the next chapter.

Chapter 5

Saliency based dictionary learning and image similarity

In literature, image classification problem is well addressed and there exist a number of sophisticated algorithms to perform the same. However majority of these methods are designed for a supervised framework, where they employ a pre-annotated training dataset. Additionally, the efficiency of these algorithms are proportional to the training data available. However, obtaining adequate data to learn a robust classifier has often proven to be difficult in several scenarios. To perform image retrieval or classification with limited training data, where the class labels cannot be exploited to learn a discrimination function, one needs to extract more informative local features from the images. Additionally, a robust similarity measure needs to be computed to compare a query with the training images.

The dictionary learning technique can be employed to provide an elegant solution to this problem in an unsupervised framework since the dictionary learning algorithm is independent of image annotations. The learned dictionaries and the corresponding sparse codes can be utilized in devising similarity measures for a pair of images [10, 11] in an unsupervised scenario. The algorithms developed for learning a dictionary generally give equal importance in reconstructing the image patches. But in an unsupervised scenario, where a robust

similarity measure needs to be computed between images, it is desired and necessary that the dictionary is learned from more meaningful image features.

To adequately exploit the limited training data in classification, we envision a saliency guided dictionary learning method and subsequently an image similarity technique for the application of classification. Our hypothesis is that extracting more meaningful features from the image is a key aspect in obtaining a more robust similarity measure between images. Some works employ image segmentation and extract features form the segmented regions. While these methods are efficient in extracting features relevant to the object, these algorithms may not be able to extract the region of interest efficiently. Moreover, the dictionary learned from an image using the relevant features, gives a compact representation of the image itself and is capable of representing images with similar content, with comparable sparse codes. Motivated by this, we design a saliency guided dictionary learning method and employ the sparse codes for computing similarity measure between a pair of images.

Objective

In this work, we propose a saliency guided dictionary learning framework where we learn a dictionary from of an image while emphasizing the reconstruction of image patches based on their saliency values. Salient object detection is exploited here to provide information regarding the importance of the candidate features. The work focuses on penalizing the local regions based on their uniqueness in an image, such that the learned dictionary provide a precise representation of salient image regions . This is advantageous in devising the similarity measure. When comparing a pair of images with similar content, the learned dictionary will represent the discriminative image features with greater accuracy and yield approximately similar sparse codes. Hence, we employ the sparse codes learned from the saliency guided dictionary to design a similarity measure.

In this work we address two main objectives:

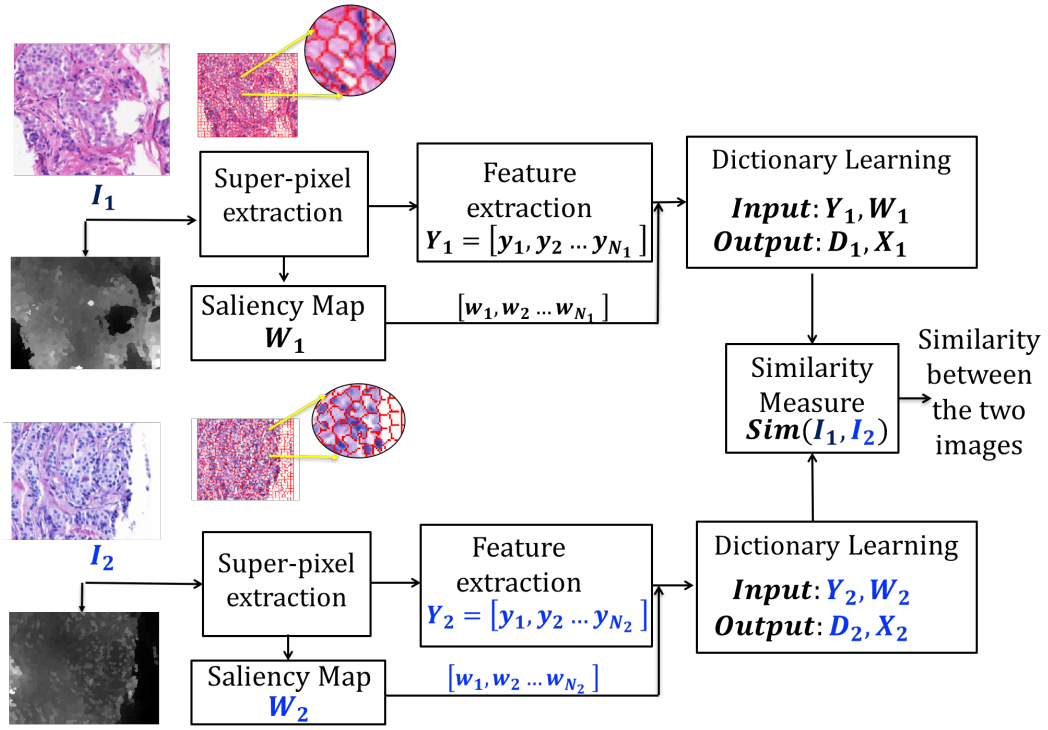


Figure 5.1: Overview of the image similarity method for images I_1 and I_2. For each image I_1 and I_2, a super-pixel segmentation is performed and an initial saliency map is obtained. The features extracted from and the corresponding saliency weights for each super-pixel are used to obtain a dictionary, update the saliency and compute the sparse codes, which are then used to evaluate similarity between the images.

a. First, we aim at designing a dictionary learning algorithm by leveraging the salient regions in an image. The objective is to generate a representative dictionary which can reconstruct the salient image regions with greater precision. The characteristic image features are manifest in the generated sparse codes.

b. Adopting the sparse representation to devise a similarity measure is the second objective of the proposed method. Here, we exploit the extent of contribution of each dictionary atom in an image representation while formulating the similarity measure. The sparse linear representation of an image with respect to the dictionary learned from another is analyzed to quantify the similarity between images.

Finally, we evaluate our algorithm in the application of histo-pathological tissue image classification. The overview of the method is shown in Fig. 5.1.

5.1 Saliency dictionary learning

The saliency guided dictionary learning method aims a obtaining a dictionary from local image regions which reconstructs salient features precisely. The local image features are reconstructed as a sparse linear combination of the dictionary by leveraging the saliency values. It should be noted here that no thresholding is performed to segment the salient regions, and hence we avoid a well known disadvantage of saliency based object detection methods: an *ad hoc* threshold selection [128]. We first propose an algorithm where the salient regions are detected and is fixed in the dictionary learning step (***SDL***). We further extend this algorithm to incorporate a spatial constraint on saliency which acts as a smoothness prior. Additionally, the saliency update is consolidated with the dictionary learning step (***SDLs***) to account for the reconstruction error.

5.1.1 SDL: Saliency based dictionary learning

The saliency map obtained from an image is used to learn the dictionary where the features from the more salient superpixel regions are given greater priority in the dictionary construction. Let $\mathbf{y}_i \in \mathbb{R}^{m\times 1}$ be a feature extracted from a local region i , the overcomplete dictionary is denoted by $\mathbf{D} \in \mathbb{R}^{m\times K}$, and the corresponding sparse codes are $\mathbf{x}_i \in \mathbb{R}^{K\times 1}$. For each region i, the normalized saliency value for the region is denoted by w_i. The saliency based dictionary is obtained by solving the following optimization.

$$\min_{\mathbf{D},\mathbf{x}} \sum_{i=1}^{N} w_i \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \|\mathbf{x}_i\|_0 \tag{5.1}$$

Here N is the total number of regions in an image. In the optimization problem given in (5.1), the saliency values of local regions are exploited as weights in the dictionary learning objective function. This facilitate that the features of the salient regions have smaller reconstruction error compared to those from the less salient regions. WIn our paper we use a contrast based salient region detection scheme to extract a probabilistic map for each of the

local regions.

$$\hat{s}_i = \sum_j \|\mathbf{f}_i - \mathbf{f}_j\|^2 \exp\left(\frac{-\|\mathbf{c}_i - \mathbf{c}_j\|^2}{2\sigma^2}\right) \tag{5.2}$$

This method of contrast based saliency detection given in eq. (5.2) captures local uniqueness of an image region. The local regions are obtained using super-pixel segmentation by employing the well known SLIC (Simple Linear Iterative Clustering) algorithm [152]. Here $\mathbf{f}_i$ is the mean color feature and $\mathbf{c}_i$ be the centroid of super-pixel i. $\hat{s}_i$ gives a measure of uniqueness of a super-pixel with respect to its neighbors and the neighborhood is determined by σ. The normalized saliency of a region is $w_i = \frac{\hat{s}_i}{\sum_i \hat{s}_i}$.

Optimization

The algorithm can be solved in a two-step approach. We first minimize eq. (5.1) to obtain the sparse codes. The following optimization is solved with fixed dictionary to obtain the sparse representation.

$$\forall i \min_{\mathbf{x}_i} \|\mathbf{x}_i\|_0 \text{ s.t. } w_i\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 \leq \epsilon \tag{5.3}$$

ϵ is a threshold on the reconstruction error. The orthogonal matching pursuit algorithm [45] is used to solve (5.3). The next step updates the dictionary using the sparse codes obtained from eq. (5.3). The dictionary is updated by solving $\min_D \sum_{i=1}^N w_i\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2$. The objective function can be re-written as,

$$\begin{aligned} & \sum_{i=1}^N \|\sqrt{w_i}\mathbf{y}_i - \sqrt{w_i}\mathbf{D}\mathbf{x}_i\|_2^2 \\ = & \sum_{i=1}^N \|\sqrt{w_i}\mathbf{y}_i - \sum_{j=i\neq k}^N \sqrt{w_i}\mathbf{d}_j x_i^j - \sqrt{w_i}\mathbf{d}_k x_i^k\|_2^2 \\ = & \sum_{i=1}^N \|(e_k)_i - \sqrt{w_i}\mathbf{d}_k x_i^k\|_2^2 = \|\mathbb{E}_k - \mathbf{d}_k X^k \mathbf{W}^{\frac{1}{2}}\|_F^2 \end{aligned} \tag{5.4}$$

Thus $\mathbf{d}_k$ is obtained by taking the singular value decomposition of $\mathbb{E}_k$=$U\Sigma V$ and $\mathbf{d}_k = U(:,1)$. It should be noted here that $\mathbb{E}_k$ is the reconstruction error of each region when not using the k^{th} atom and are weighted by the saliency of the regions. $\mathbf{E_k}$ defined in 2.5, is the error weighted uniformly. The region saliency values, used in this manner, emphasize the data points which should contribute more in updating the dictionary atoms. $\mathbf{X}^k$ is the k^{th} row of $\mathbf{X} \in \mathbb{R}^{K \times N}$, $\mathbf{W}$ is a diagonal matrix with diagonal entry $\mathbf{W}(i,i) = w_i$.

Algorithm 2 Algorithm SDL

Input: Superpixels i, $\mathbf{f}_i$, c_i, $\mathbf{y}_i$ $\forall i$
Output:w_i, $\mathbf{D}$, $\mathbf{X}$
For time $t = 0$

Initialize $\mathbf{D}_0$: The initialization of $\mathbf{D}_0$ is done by selecting top K salient data points

Compute saliency map: $\hat{s}_i$ using (5.2) and obtaining the normalized saliency weights by $w_i = \frac{\hat{s}_i}{\sum_i \hat{s}_i}$

For time $t > 0$ until convergence (or a fixed number of iterations)

Sparse code update: While keeping the dictionary fixed, update $\mathbf{x}_i$ using (5.3).

Dictionary update: Keeping the sparse code fixed update each column of the dictionary by using (5.4) and solving

$$\min_{\mathbf{D}} \sum_{i=1}^{N} w_i \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2$$
$$\min_{\mathbf{d}_k} \|\mathbb{E}_k - \mathbf{d}_k \mathbf{X}^k \mathbf{W}^{\frac{1}{2}}\|_F^2 \quad \forall k$$

$\mathbf{d}_k$ is obtained by taking the singular value decomposition of $\mathbb{E}_k$=$\mathbf{U\Sigma V}$ and $\mathbf{d}_k = \mathbf{U}(:,1)$. $\mathbf{X}^k$ is the k^{th} row of $\mathbf{X} \in \mathbb{R}^{K \times N}$

5.1.2 SDLs: Saliency based dictionary learning with spatial constraint

In SDL described in Section 5.1.1, we design a dictionary learning scheme where we learn a dictionary with the reconstruction error weighted by the region saliency values. In the optimization process, the saliency values are fixed to the initial detection.

In this section we design an algorithm, where the saliency values get adaptively updated at each iteration, as we learn the dictionary. The saliency map calculated in sec. 5.1.1 was based on contrast in a local neighborhood. This often leads to a non-smooth map, which is higher near the object boundaries and non-homogeneous inside the object regions.

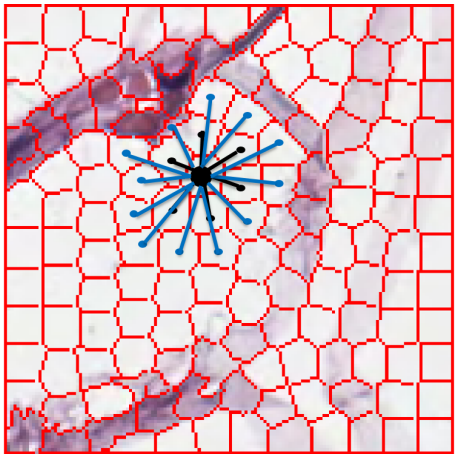


Figure 5.2: The figure shows the neighborhood selection of a super-pixel. The neighboring super-pixels (denoted by small *black* dots), which have a common boundary portion and second order neighbor (denoted by small *blue* dots) are considered to be in the neighborhood of the superpixel (denoted by larger circle). The lines between main superpixel and its neighboring superpixels denote the presence of an edge between the regions.

This non-smooth nature of the saliency map can be observed from the second row of Fig. 5.3. To account for this we design a new algorithm, where we constrain the saliency map by a spatial smoothness function. Additionally, the saliency map is updated in conjunction with the dictionary to take into account, that regions with high reconstruction error should not have high saliency values. The SDLs algorithm is formulated by the following optimization.

$$\min_{\mathbf{D},\mathbf{X},\mathbf{W}} \quad \sum_{i=1}^{N} w_i \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \sum_{i,j} a_{ij} \|w_i - w_j\|_2^2 + \mu \sum_{i}^{N} \tilde{d}_{ii} \|w_i - \frac{w_0}{\tilde{d}_{ii}}\|_2^2 + \|\mathbf{x}_i\|_0 \quad (5.5)$$

The weights between nodes are assigned as

$$a_{ij} \begin{cases} = e^{-\|\mathbf{f}_i - \mathbf{f}_j\|^2} & \text{if } j \in \mathcal{N}_i \\ = 0 & \text{otherwise} \end{cases} \quad (5.6)$$

Here $\mathcal{N}_i$ is the neighborhood of super-pixel i. The neighborhood is denoted by the super-pixels

that share a part of their boundary with that of i as shown in Fig. 5.2. $\tilde{\mathbf{D}}$ is a a diagonal matrix with diagonal entries, $\tilde{d_{ii}} = \sum_{j=1}^{N} a_{ij}$. The spatial smoothness constraint is imposed by $\sum_{i,j} a_{ij}\|w_i - w_j\|_2^2$, such that in a local neighborhood if there is a strong connectivity, the saliency map should be homogeneous. The term $\sum_i^N \tilde{d_{ii}}\|w_i - \frac{w_0}{\tilde{d_{ii}}}\|_2^2$ restrain the saliency values to the previously detected saliency.

Optimization

Since the objective function is non-convex with respect to $\mathbf{W}$, $\mathbf{D}$ and $\mathbf{X}$, the variables are updated by an alternative minimization process.

i. Update $\mathbf{W}$: For the update step pf saliency values, the dictionary and the sparse codes are kept fixed. To obtain $\mathbf{W}$, we differentiate the cost function with respect to each w_i and set it to 0. The optimization can be solved by writing it in vector notation as follows,

$$\begin{aligned}\frac{\partial C}{\partial \mathbf{w}} &= \frac{\partial}{\partial \mathbf{w}}\left\{\sum_{i=1}^{N} w_i\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \sum_{i,j} a_{ij}\|w_i - w_j\|_2^2 + \frac{1}{2}\mu\sum_{i}^{N}\tilde{d_{ii}}\|w_i - \frac{w_0}{\tilde{d_{ii}}}\|_2^2\right\} \\ &= \frac{\partial}{\partial \mathbf{w}}\left\{\mathbf{E}^T\mathbf{w} + \frac{1}{2}\mathbf{w}^T(\tilde{\mathbf{D}} - \mathbf{A})\mathbf{w} + \frac{1}{2}\mu(\mathbf{w} - \tilde{\mathbf{D}}^{-1}\mathbf{w}_0)^T\tilde{\mathbf{D}}(\mathbf{w} - \tilde{\mathbf{D}}^{-1}\mathbf{w}_0)\right\} \quad (5.7)\end{aligned}$$

setting $\frac{\partial C}{\partial \mathbf{w}}\big|_{\mathbf{w}^*} = 0$, a closed form solution of $w \in \mathbb{R}^{N\times 1}$ can be obtained as

$$\mathbf{w}^* = \beta^{-1}(\tilde{\mathbf{D}} - \alpha\mathbf{A})^{-1}(\mu\mathbf{w}_0 - \mathbf{E}). \quad (5.8)$$

The details of the derivation is given in appendix C.1 Here $\mathbf{E} = [\|\mathbf{y}_1 - \mathbf{D}\mathbf{x}_1\|_2, \|\mathbf{y}_2 - \mathbf{D}\mathbf{x}_2\|_2 \ldots \|\mathbf{y}_N - \mathbf{D}\mathbf{x}_N\|_2]^T$. α, β are functions of μ, such that $\alpha = \frac{1}{1+\mu}$ and $\beta = 1 + \mu$. Since for an image, $\tilde{\mathbf{D}}$ and $\mathbf{A}$ is fixed, the matrix inversion in eq. (5.8) needs to be computed only once.

ii. Update $\mathbf{D}$: After updating the saliency weights, the dictionary is updated next. The dictionary update is done in the similar manner as in (5.4) while keeping the $\mathbf{W}$ and $\mathbf{X}$ fixed

iii. Update $\mathbf{X}$: the update of $\mathbf{X}$ is also done in the similar manner as it is done in *SDL*. While keeping $\mathbf{W}$ and $\mathbf{D}$ fixed, $\mathbf{X}$ is obtained by solving (5.3).

Algorithm 3 Algorithm SDLs

Input: Superpixels i, $\mathbf{f}_i$, c_i, $\mathbf{y}_i$ $\forall i$
Output:$\mathbf{W}$, $\mathbf{D}$, $\mathbf{X}$
For time $t = 0$

Initialization: $\mathbf{D}_0$ is initialized by selecting random patches from the image. Initialize $\mathbf{W}_0$ using normalized saliency values obtained from eq. (5.2). Compute the matrix inverse in eq. (5.8).

For time $t > 0$ until convergence or fixed number of iterations

Sparse code update: While keeping the dictionary and the saliency weights fixed, updtae x_i using (5.3). The $\mathbf{E}$ is computed using the dictionary and the corresponding updated sparse codes.

Dictionary update: Keeping the sparse code fixed update each column of the dictionary by using (5.4) and solving (6.11).

Update saliency map: The saliency map $\mathbf{W}_t$ for iteration t is obtained by solving the closed form solution presented in (5.8). Set $\mathbf{W}_0 = \mathbf{W}_t$. The update for the saliency is performed till the condition $\mathbf{W}_t - \mathbf{W}_{t-1} \leq \tau$ is reached.

A compact algorithmic description for *SDL* and *SDLs* is shown in algorithm 2 and 3 respectively.

5.2 Image similarity using sparse codes

The saliency incorporated dictionary learning provides us with compact sparse codes, which can further be analyzed in comparing images. To obtain the similarity between a pair of images, we exploit the sparse codes in designing a histogram for the images. These features can be compared using any histogram comparing methods e.g., K-L divergence, chi-square measure, histogram bin ratio, *etc.* or other distance measures to obtain the similarity between images [137, 141]. As demonstrated [10, 11, 143, 154], compressibility of the sparse representation can be exploited in computing similarity between images.

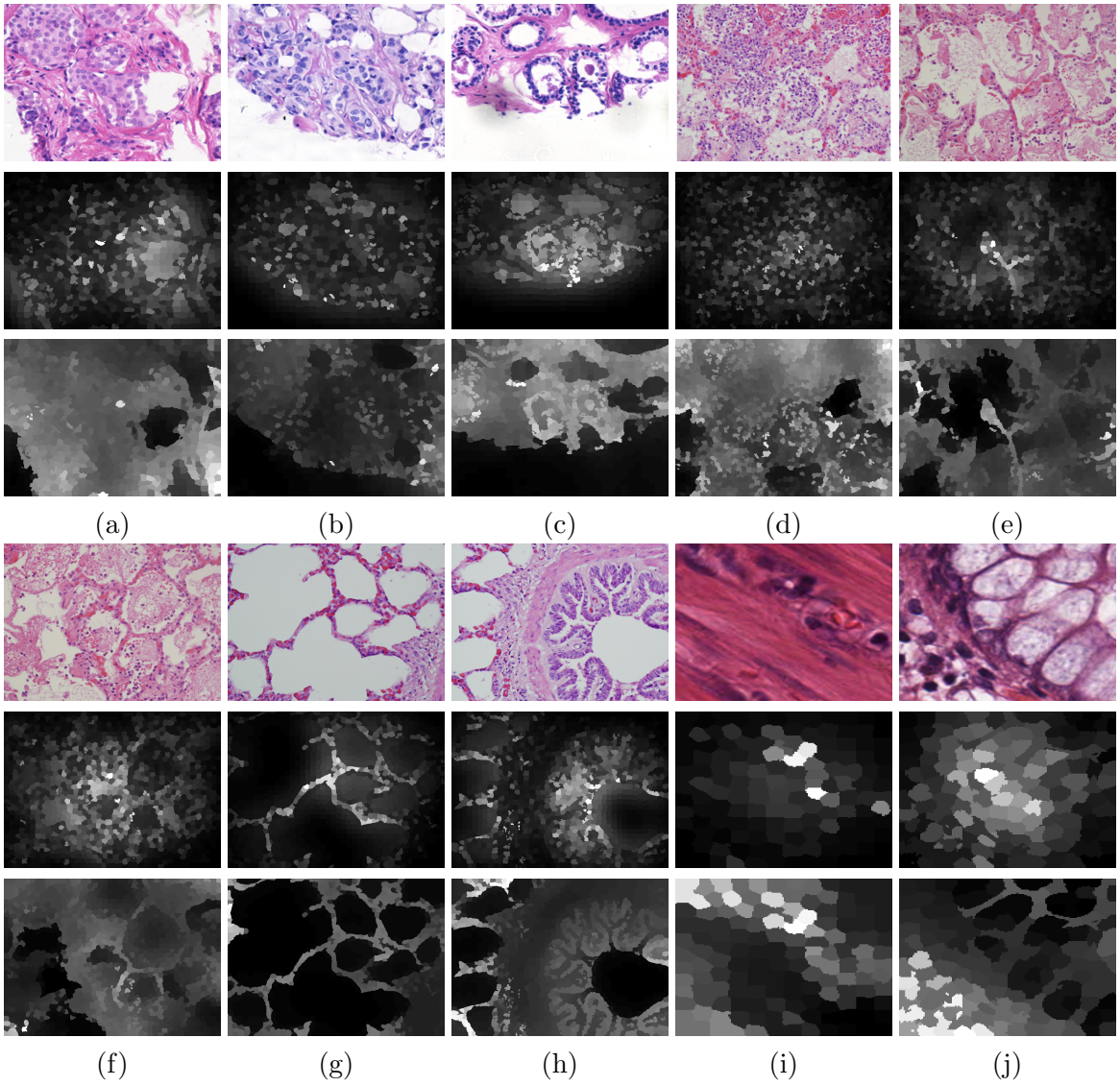


Figure 5.3: The first row of the figure shows the original images. The second and third rows show the saliency maps computed by eq. (5.2) and the final maps when updated using SDLs respectively. Images (a-c) show samples from the Bisque dataset containing breast cancer tissue images. (d-h) shows an image from the ADL tissue dataset. (i) and (j) show samples from the colorectal cancer tissue dataset.

5.2.1 Cross dictionary representation

Let $\mathbf{I}_1$ and $\mathbf{I}_2$ be the two images for which similarity is to be computed. $\mathbf{Y}_1 = [\mathbf{y}_1, \ldots, \mathbf{y}_{N_1}]$ and $\mathbf{Y}_2 = [\mathbf{y}_1, \ldots, \mathbf{y}_{N_2}]$ are the features obtained from the two images where N_1 and N_2 are the number of super-pixels in the images respectively. $\mathbf{W}_1 \in \mathbb{R}^{N_1 \times N_1}$ and $\mathbf{W}_2 \in \mathbb{R}^{N_2 \times N_2}$ are the matrices containing the saliency values of $\mathbf{I}_1$ and $\mathbf{I}_2$ along the diagonal corresponding to

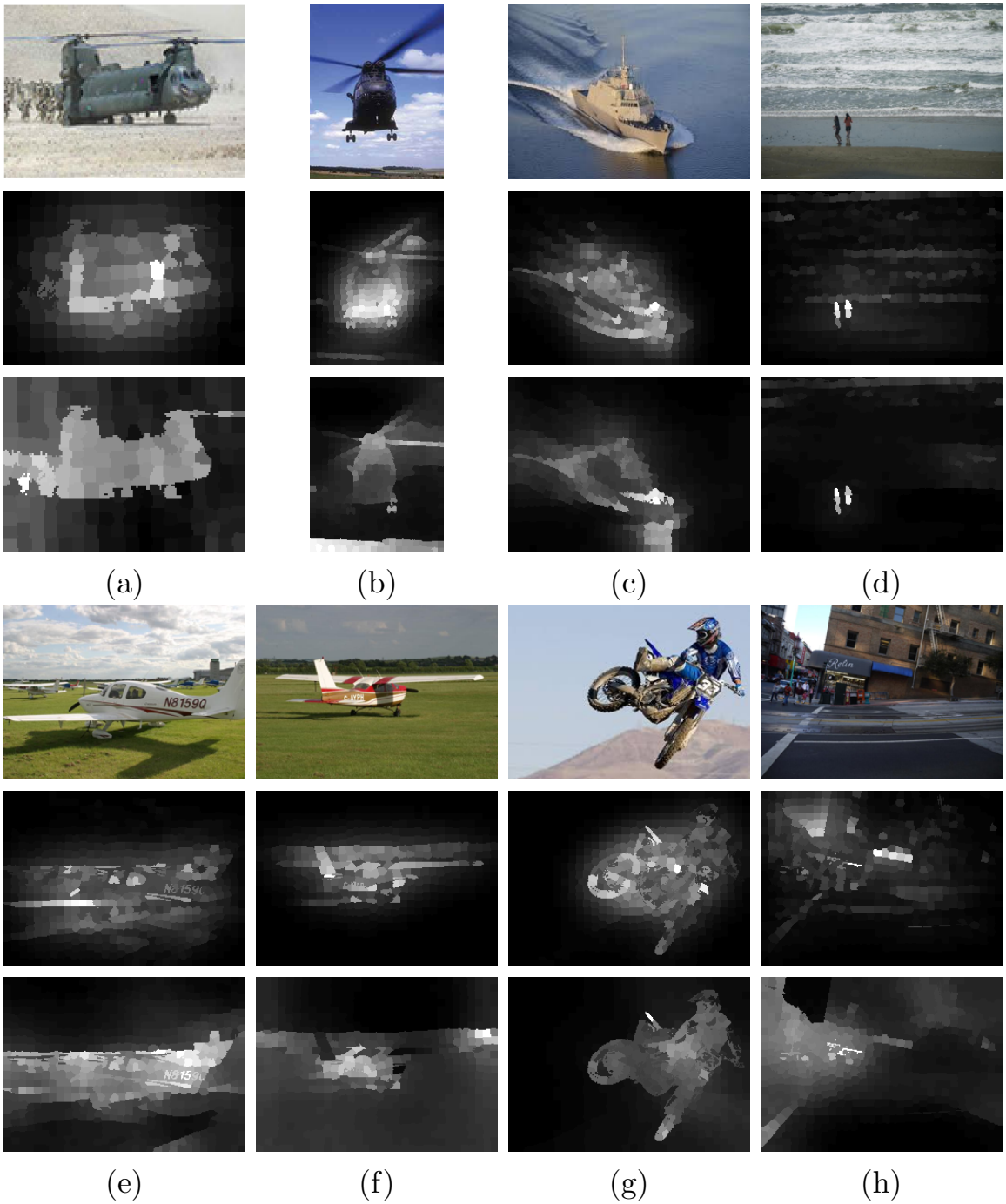


Figure 5.4: The first row of the figure shows the original images. The second and third rows show the saliency maps computed by eq. (5.2) and the final maps when updated using SDLs respectively. The images are from the datasets used in [10] and [153].

each super-pixel i. Let $\mathbf{D}_1$ and $\mathbf{D}_2$ be the dictionaries learned from the images $\mathbf{Y}_1$ and $\mathbf{Y}_2$ respectively and $\mathbf{X}_1 = [\mathbf{x}_1, \ldots, \mathbf{x}_{N_1}]$ and $\mathbf{X}_2 = [\mathbf{x}_1, \ldots, \mathbf{x}_{N_2}]$ be the corresponding *self sparse*

codes obtained by solving the following

$$\begin{aligned}\forall i \in 1...N_1 \min_{\mathbf{x}_i} \|\mathbf{x}_i\|_0 \text{ s.t.} \|(\mathbf{Y}_1 - \mathbf{D}_1\mathbf{X}_1)\mathbf{W}_1^{\frac{1}{2}}\|_F^2 \leq \epsilon \\ \forall i \in 1...N_1 \min_{\mathbf{x}_i} \|\mathbf{x}_i\|_0 \text{ s.t.} \|(\mathbf{Y}_2 - \mathbf{D}_2\mathbf{X}_2)\mathbf{W}_2^{\frac{1}{2}}\|_F^2 \leq \epsilon \end{aligned} \tag{5.9}$$

$\mathbf{D}_1$ and $\mathbf{D}_2$ are considered as the universal encoder that provide a compact representation $\mathbf{X}_1$ and $\mathbf{X}_2$ for images $\mathbf{I}_1$ and $\mathbf{I}_2$ respectively. When $\mathbf{I}_1$ and $\mathbf{I}_2$ are represented with respect to $\mathbf{D}_2$ and $\mathbf{D}_1$, the *relative sparse codes* $\mathbf{X}_{1|2}$ and $\mathbf{X}_{2|1}$ are obtained using the following,

$$\begin{aligned}\forall i \in 1...N_2 \min_{\mathbf{x}_i} \|\mathbf{x}_i\|_0 \text{ s.t.} \|(\mathbf{Y}_1 - \mathbf{D}_2 X_{1|2})\mathbf{W}_1^{\frac{1}{2}}\|_F^2 \\ \forall i \in 1...N_2 \min_{\mathbf{x}_i} \|\mathbf{x}_i\|_0 \text{ s.t.} \|(\mathbf{Y}_2 - \mathbf{D}_1\mathbf{X}_{2|1})\mathbf{W}_2^{\frac{1}{2}}\|_F^2 \end{aligned} \tag{5.10}$$

Where $\mathbf{x}_i$ denote a column of $\mathbf{X}$.

5.2.2 Similarity measure using codelength overhead

It has been shown in [11,142,146,147] that compressibility of an image when represented with respect to information available from a second image, can be exploited to obtain a similarity measure between the two images. As stated in [155], the code length of a datum gives another way to represent its probability distribution. If we consider y as the data to be encoded by encoder D_i (the dictionary learned from the data can be assumed as the universal encoder for that data, which was also mentioned in [11]), the code obtained is x and *l(x)* is the length of the codeword x, then there exists a sub-probability mass function for x, $P(x) = 2^{-l(x)}$ ([155]). Moreover it was shown in [155] that if $P_1(x)$ denotes the sought-after universal representation of x and $P_i(x)$ is any other representation of the data x, then the quantity of interest to be minimized in finding the universal representation is the difference between the codelengths

generated by two different encoders, given by,

$$\log \frac{1}{P_1(x)} - \log \frac{1}{P_i(x)}$$

This term is often called as codelength overhead. While this quantity can provide an insight in choosing the optimal encoder (or the distribution) to represent the data, we exploit this concept of codelength overhead in evaluating the similarity between two data (image in this case). Consider x_1 and x_2 be sparse codes of the two images whose similarity needs to be measured and θ_1 and θ_1 are the encoders that provide most compressed representation of the images. Here, by universal representation, we seek the most compressible representation of x. In this case we assume that the universal representation as well as the encoder providing the representation is available in the form of the learned dictionary and the sparse codes, respectively. The purpose is to evaluate how well their encoders can represent each other. Hence, we can say that the codelength overhead between the two data, when encoded with the other's universal encoder, can provide a measure for similarity between the data, given as

$$\mathbf{s}(x_1, x_2) = \max_{\theta \in \theta_1, \theta_2} | \log \frac{1}{P_\theta(x_1)} - \log \frac{1}{P_\theta(x_2)} | \tag{5.11}$$

The maximum value of the two differences provides the worst case codelength difference between the two images. It also emphasizes the idea that if two images have similar content, the codes obtained when represented with each others encoders will achieve similar compressibility.

Properties of $\boldsymbol{s}(x_1, x_2)$

1. *Positivity*: $\mathbf{s}(x_1, x_2) \geq 0$, since it is defined as the absolute value of the codelength difference, the quantity $\mathbf{s}(x_1, x_2)$ is always greater that 0.

2. *Symmetricity*: $\mathbf{s}(x_1, x_2) = \mathbf{s}(x_2, x_1)$, as the codelength difference is obtained when represented with respect to each other's encoders, the difference remains the same, as

well the maximum value of the difference.

3. $\mathbf{s}(x_1, x_2) = 0$, when $x_1 = x_2$.

Let $\theta_1 = D_1$ and $\theta_2 = D_2$, then $\log \frac{1}{P_{D_1}(X_1)} = \frac{\|X_1\|_0}{N_1}$, $\log \frac{1}{P_{D_2}(X_1)} = \frac{\|X_{1|2}\|_0}{N_1}$, $\log \frac{1}{P_{D_2}(X_2)} = \frac{\|X_2\|_0}{N_2}$, $\log \frac{1}{P_{D_1}(X_2)} = \frac{\|X_{2|1}\|_0}{N_2}$. The similarity between I_1 and I_2 can be obtained by using the similarity measure explained in sec. 5.2.2,

$$\mathbf{s}(X_1, X_2) = \max_{D \in D_1, D_2} | \log \frac{1}{P_D(X_1)} - \log \frac{1}{P_D(X_2)} |$$

Since the dictionary learned from the images takes into account the saliency of the image regions, images with similar content will have a comparable compressed representation with respect to each others dictionary.

For image classification, the codelength of the query image is compared to that of all the images in the dataset, and the one with the minimum codelength overhead is chosen as the best matching image which is same as the 1 nearest neighbor classifier, given by

$$\min_{X_m, m \in 1..M} \max_{D \in D_1, D_m} | \log \frac{1}{P_D(X_1)} - \log \frac{1}{P_D(X_m)} | \tag{5.12}$$

where M denotes the number of images in the dataset.

5.2.3 Histogram using compression of sparse codes

In this work, we propose a similarity measure that exploits histograms using the sparse codes x_i, which can be used in conjunction with any histogram based similarity measure. To design the histogram, we use this compressibility of the codes while taking into account the contribution of each dictionary atom [21]. The basic idea presented here for designing the similarity between images is to represent the feature of one image with respect to another and compare how well their respective dictionaries can represent one another. If I is an image and X the sparse code representation of local image features, the sparse code histogram is

computed as

$$\mathbf{h}_I(k) = \|\mathbf{X}^k\|_0 \tag{5.13}$$

where $\mathbf{X}^k$ is the k^{th} row of the matrix $\mathbf{X}$. The dictionary atoms act as the bin centers of the histogram and the number of features that share the dictionary atom constitute the bin frequency. Construction of such histogram gives an idea, how frequently one particular dictionary atom is being used in representing the image features. The histogram is normalized by the total number of super-pixel features. We denote the histogram of the *self sparse codes* as :

$$\mathbf{h}_1(k) = \|\mathbf{X}_1^k\|_0; \;\; \mathbf{h}_2(k) = \|\mathbf{X}_2^k\|_0 \tag{5.14}$$

While the *relative sparse code* histograms can be writen as:

$$\mathbf{h}_{1|2}(k) = \|\mathbf{X}_{1|2}^k\|_0; \;\; \mathbf{h}_{2|1}(k) = \|\mathbf{X}_{2|1}^k\|_0 \tag{5.15}$$

Each histogram bin accounts for the frequency of occurrence of a dictionary atom in representing an image feature. While the histogram from the *self sparse codes*, gives the frequency of occurrence of the atoms in an images, the *relative sparse code* histograms shown the frequency of occurrence when the images are represented with each others dictionary.

When two images are significantly different from one another with respect to their content, the change in the actual sparse code histograms $\mathbf{h}_1$, $\mathbf{h}_2$ as well as the *relative sparse code histograms* $\mathbf{h}_{1|2}$, $\mathbf{h}_{2|1}$ will also be significant. This change can be quantified by the Kullback−Leibler (KL) divergence [138] method, which is a measure to compute difference between two discrete probability distribution function $\mathbf{p}$ and $\mathbf{q}$ and is given as,

$$KL(\mathbf{p}||\mathbf{q}) = \sum_i \mathbf{p}(i) \log \frac{\mathbf{p}(i)}{\mathbf{q}(i)} \tag{5.16}$$

In particular, we compute the final KL divergence as an aggregate of the KL divergence of histogram pairs,

$$\mathcal{S}(\mathbf{I}_1, \mathbf{I}_2) = KL(\mathbf{h}_1||\mathbf{h}_{2|1}) + KL(\mathbf{h}_{2|1}||\mathbf{h}_1) + KL(\mathbf{h}_2||\mathbf{h}_{1|2}) + KL(\mathbf{h}_{1|2}||\mathbf{h}_2) \tag{5.17}$$

The similarity measure developed here follows the following property,

1. *Symmetry* i.e., $\mathcal{S}(\mathbf{I}_1, \mathbf{I}_2) = \mathcal{S}(\mathbf{I}_2, \mathbf{I}_1)$. The KL divergence $KL(\mathbf{p}||\mathbf{q})$ itself is not symmetric but, $KL(\mathbf{p}||\mathbf{q}) + KL(\mathbf{q}||\mathbf{p})$ is symmetric.

2. *Positivity* i.e., $\mathcal{S}(\mathbf{I}_1, \mathbf{I}_2) \geq 0$ The KL divergence between two normalized histograms always greater than 0. In eq. (5.17), each of the four parts have equal contribution, thus $\mathcal{S}(\mathbf{I}_1, \mathbf{I}_2)$ is also greater than 0

We perform k nearest neighbor [156] search using the K-L divergence of the sparse code histograms for image classification. The images in the dataset showing minimum divergence, $\mathcal{S}$ is given as the best match. For a dataset containing $p = 1 \ldots P$ images and q is a test (query image), the best match is given by the following

$$\min_p \mathcal{S}(\mathbf{I}_p, \mathbf{I}_q) \tag{5.18}$$

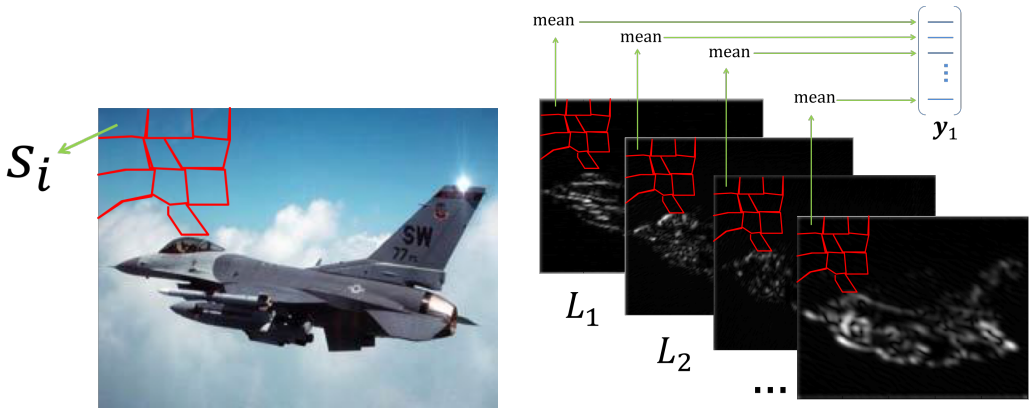
5.3 Experiments

Sparse representation and dictionary learning have been efficiently used in various image classification methods [12, 47, 65, 91]. However, all these methods incorporate the information of the image category to design a classifier while learning a sparse representation of the

features. In certain applications, specifically histo-pathological tissue classification, manual labeling of large datasets are often expensive and training dataset is limited. Here we perform image classification for histo-pathological tissue image retrieval and military vehicle type identification. In both applications, large annotated datasets are not readily available. Another challenge is the imbalance in the various categories of the training dataset. In practical scenarios in tissue image classification for e.g., the number of images available for healthy tissue might be significantly less than that of diseased tissues. In such scenarios, designing a classifier which would detect the class boundaries efficiently becomes a challenging task. In the following subsections, we demonstrate the validation of our proposed method. We first describe the image features used in our experiments, the comparison algorithms and finally describe in details the three different tissue datasets and the experimental results.

5.3.1 Image feature

In sparse representation based classification using local features, both image intensities or other discriminative image features have been exploited. However since in our experiments, the local image regions are defined by the super-pixels of different size, using just image intensity lead to different size feature vectors. Moreover, image intensity or color histograms are not sufficient to discriminate between images, specially in case of Hematoxylin and Eosin (H&E) histo-pathological tissue images. In [157] it was demonstrated that texture features can efficiently discriminate between different tissue types. Also in case of differentiating between military camouflage uniform patterns or discriminating between military and civil vehicles, texture patterns play a key role. In our experiments we employ local Gabor features [24, 158] to represent the super-pixels. In the literature, it has been shown that Gabor filters can approximate the characteristics of certain cells in the mammalian visual cortex and can be exploited in getting the texture information of an image. 2D Gabor filters are obtained by combining Gaussian kernel with sinusoidal functions of different frequency and orientation.



(a) (b)

Figure 5.5: (a) shows the superpixel construction for an image. (b) shows the different Gabor filter response $L_1 \ldots L_{32}$ and how the final local feature $\mathbf{y}_i$ is obtained by taking the spatial maximum for a superpixel at each level of response

The Gabor filters are regulated by the standard deviation of the Gaussian filters and the orientation of the sinusoidal functions. An image is convolved with different combinations of the standard deviation and the frequency to get the Gabor filter response. The mean response of each filter bank can be used as a global texture feature of the image. For a local region, we first convolve the image with 32 Gabor filters, for 8 orientations and 4 different Gaussian filters. We then compute the maximum response within a superpixel for each of the 32 filters and thus obtain a 32 dimensional Gabor feature for each region as shown in Fig. 5.5. To account for the variation in intensity, we also include local color features in our descriptor. For each super-pixel, we compute the mean color of the region in CIE L^*a^*b color space along each channel. We concatenate the color and Gabor feature of a region to get the final local descriptor.

5.3.2 Performance evaluation

We compare our algorithm with four different methods from literature. First, we compare with the sparse representation based classification [15] scheme. Second we compare with the method of spatial pyramid matching [139], a scheme for aggregating local features while keeping their spatial relation. The third and fourth methods are based of compression based

similarity measure [10, 11]. We also show comparison of the designed similarity measure without incorporating the saliency.

i. *Sparse representation based classification* (SRC): In [15] is was shown that an image of a specific category can be represented as a linear combination of other images in the dataset. It was shown in [15], in face image retrieval, that the face image of a subject could be represented as a linear combination of that subjects face images acquired under various lighting conditions. A test image is represented as a sparse combination of these basis for each of the category present and classified based on the minimum reconstruction error. In our comparisons, we apply SRC on the vectorized gray-scale images. Let $[\mathbf{D}_1, \mathbf{D}_2, ..., \mathbf{D}_k]$ denote the dictionary, where D_i is the sub-dictionary created by stacking the vectorized gray-scale images of class i as columns of the dictionary. The sparse code $\mathbf{x}$ of a test image $\mathbf{y}$ is obtained by minimizing the equation

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2 + \gamma\|\mathbf{x}\|_1 \tag{5.19}$$

The classification is done based on minimum reconstruction error using each sub-dictionary given as follows

$$\min_{i} \|\mathbf{y} - \mathbf{D}_i\mathbf{x}_i\|_2 \tag{5.20}$$

Here $\mathbf{x}_i$ is the elements of vector $\mathbf{x}$ corresponding to the sub-dictionary $\mathbf{D}_i$. We use the CVX (*http://cvxr.com/cvx/*) package to compute the sparse representation of the images.

ii. *Spatial pyramid matching* (SPM):The method described in [139] combines local image features while retaining the spatial correspondence. In this method an image is partitioned into subregions, and for each subregion a histogram of local features is computed. Finally the histograms from each subregion are concatenated to obtaining the final feature representation. For our experiments we use the same local image features as described in Sec. 5.3.1 and two pyramid levels. We compare the spatial pyramid histograms using

KL divergence, such that if $\mathbf{p}$ and $\mathbf{q}$ are the spatial pyramid histograms of two images, then the KL divergence is given as:

$$KL(\mathbf{p}||\mathbf{q}) = \sum_i \mathbf{p}(i) \log \frac{\mathbf{p}(i)}{\mathbf{q}(i)} + \sum_i \mathbf{q}(i) \log \frac{\mathbf{q}(i)}{\mathbf{p}(i)} \tag{5.21}$$

The test image is classified using a nearest neighbor classifier given by equation (5.18).

iii. *Sparse representation based compression distance* (SCD): In [11], the authors developed a compression based similarity measure. Here, a dictionary and corresponding sparse codes are learned for each image in the dataset. While computing the similarity, a pair of images are represented with respect to each other's dictionary. The compressibility of these sparse codes are exploited in computing a similarity measure.

iv. *Saliency guided dictionary and sparse code compression distance* (SLIDE, SLIDEs, SLIDEdl): In [10], the local image features were represented based on a learned dictionary by leveraging the salient region features, similar to SDL presented in the paper. While the similarity measure is based on code length overhead, where the code length is defined by the ℓ_0 norm of the sparse representation. The overhead for a pair of images is given by the difference in the code lengths when represented with the dictionary of another image. *SLIDE* is based on the idea that two images with similar content will be represented more compactly with each other's dictionaries. We refer *SLIDE* and *SLIDEs* when we use the *SDL* and *SDLs* algorithm in conjunction with the compression distance, whereas we *SLIDEdl* refers to using the sparse code compression based distance without saliency detection.

v. *Saliency guided dictionary and sparse code sparse code histogram*(SDL,SDLs,DL+KLdiv): When we refer *SDL* and *SDLs*, we denote the two algorithms in combination with sparse code histogram and K-L divergence for similarity evaluation. We also show results using dictionary learning without leveraging the saliency values. Here the dictionary is learned

Table 5.1: Confusion matrix % for ADL dataset

Class		Kidney		Lung		Spleen	
	Method	Inflamed	Normal	Inflamed	Normal	Inflamed	Normal
Inflamed	*SDL*	**97.8**	2.2	97.4	2.6	95	5
	SDLs	**95.6**	4.4	**100**	0	95	5
	DL+KLdiv	93.3	6.7	97.4	2.6	**97.5**	2.5
	SLIDE [10]	24.5	75.5	92.3	7.7	57.5	42.5
	SCD [11]	82.3	17.7	41	59	65	35
	SRC** [15]	71.1	28.9	43.8	56.4	67.5	32.5
	SPM [139]	86.7	13.33	94.9	5.13	87.5	12.5
	SHIRC* [14]	83.1	16.9	71.0	29	69.4	30.6
	DFDL* [16]	90.0	10	97.4	2.6	92.0	8
Normal	*SDL*	12.5	87.5	2.56	**97.4**	9.8	90.2
	SDLs	10	90	2.56	**97.4**	7.3	**92.7**
	DL+KLdiv	10	90	5.1	94.9	9.8	90.2
	SLIDE [10]	15	85	67.3	32.7	56.1	43.9
	SCD [11]	73	27	35.9	64.1	60.8	39.2
	SRC** [15]	20	80.0	28.2	71.8	43.9	56.1
	SPM [139]	10.4	89.5	10.8	89.2	23.1	76.9
	SHIRC* [14]	7.9	**92.1**	9	91.0	9.2	90.8
	DFDL* [16]	11.8	88.2	3.5	96.5	7.1	**92.9**

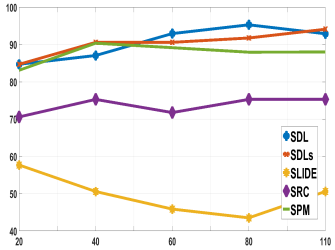
*Results as presented by the authors in the paper.

**Experiments with full images.

by solving the optimization problem given in (2.2) using the super-pixel Gabor features for each image. The similarity between image is performed using the KL-divergence of the sparse code histograms as described in Sec. 5.2 and the similarity measure given in (5.17)

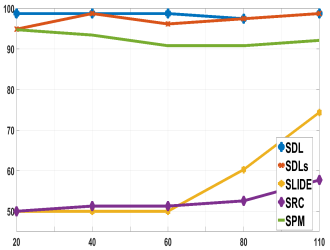
5.3.3 Application to tissue image classification

In tissue images, saliency detection helps in accentuating the structures that distinguish between healthy and diseased cells or differentiate one tissue type from another. For example, in differentiation between malignant and benign tumor, often the smoothness of the nuclei are taken into account [159]. While in other applications there different types to tissues are required to be classified [160] based on their textural patterns to determine cellular composition.For example, in Fig. 5.3 (e) and (f), the samples from a colorectal tissue image dataset, the saliency maps make the structural characteristics of the images more prominent. We demonstrate two applications: first to distinguish between malignant and benign tissues,



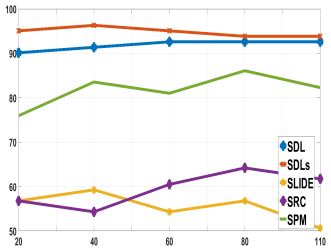
RMSD inflamed (%): SDL = **3.3**; SDLs = 7.2; SLIDE = 10.6; SRC = 7.4; SPM = 3.7
RMSD normal (%): SDL = 8.2; SDLs = **4.2**; SLIDE = 19.9 ; SRC = 4.5; SPM = 4.3

(a) Kidney



RMSD inflamed (%): SDL = 1.3; SDLs = 5; SLIDE = 3.8; SRC = 9.0; SPM = **0**
RMSD normal (%): SDL = **1.8**; SDLs = 3.4; SLIDE = 9.2; SRC = 8.9; SPM = 3.3

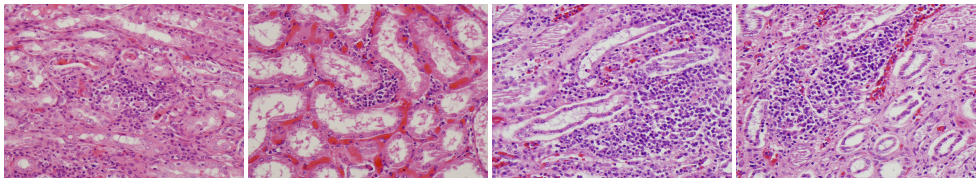
(b) Lung



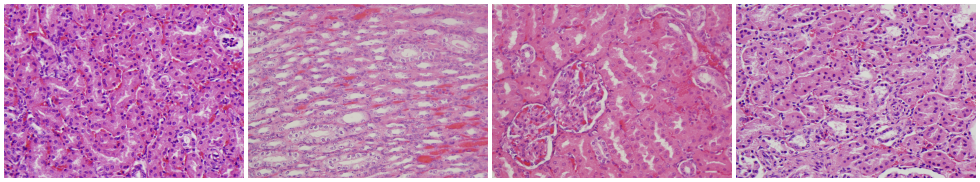
RMSD inflamed (%): SDL = **1.3**; SDLs = 2.2 SLIDE = 11.3; SRC = 2.2; SPM = 2.5
RMSD normal (%): SDL = **2.7**; SDLs = 3.8; SLIDE = 6.7; SRC = 11.6; SPM = 8.6

(c) Spleen

Figure 5.6: Plot showing classification accuracy (%) with changing size of training dataset for five different methods SDL, SDLs, SLIDE [10], SPM [139] and SRC [15]. The x-axis denotes the number of training images used per class while the classification accuracy % is plotted along the y-axis. Below each figure the RMSD for the two classes are given.



Kidney inflammation



Kidney normal

Figure 5.7: ADL tissue examples of kidney.

as a binary classification problem, and second differentiate between tissue types as a multiclass classification problem. Three different datasets have been used in this experiment. We provide a detailed description of the three datasets below. For all the three datasets, we compute a combination of local Gabor and color features as discussed in sec. 5.3.1, to learn the dictionary and sparse codes.

For all our experiments we use $\alpha = 0.9$. A value of α close to 1 preserves the neighborhood graph structure . For SDLs, at each iteration step, a normalization needs to be performed. Although the error ϵ are restricted to 10^{-5} or below, there can be scenarios where $(\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2)$ is greater that w_i in eq. 5.8. To restrict the degeneration of updated saliency values, we

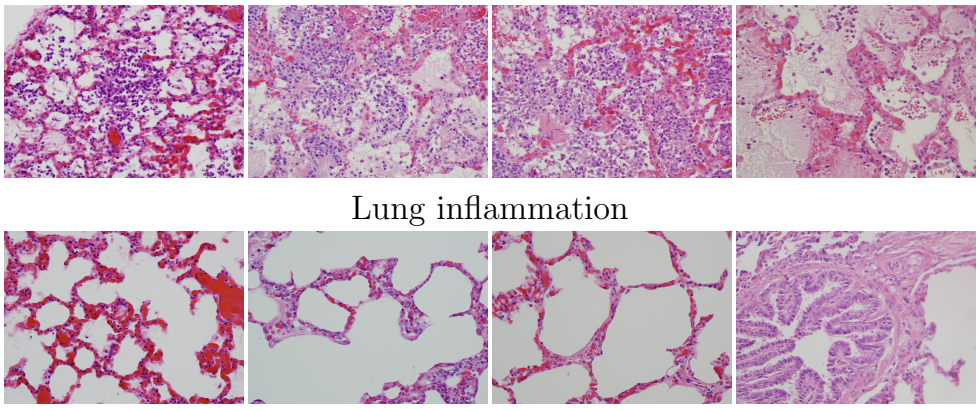


Figure 5.8: ADL tissue examples of lung.

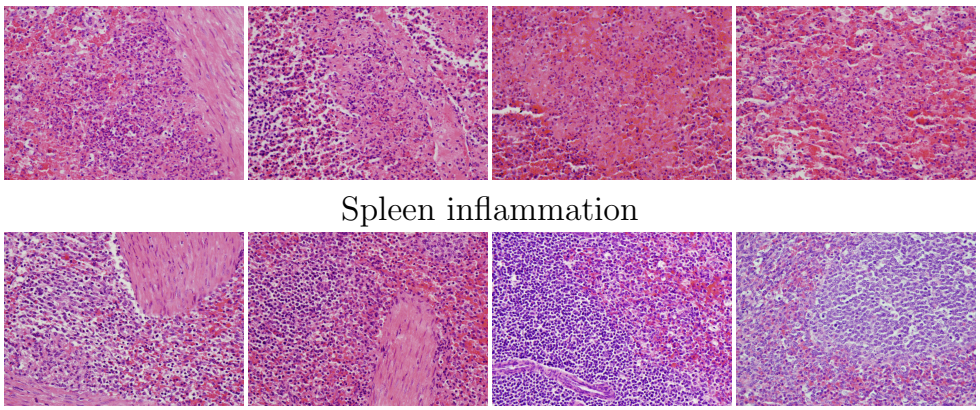


Figure 5.9: ADL tissue examples of spleen.

normalize $(\mu \mathbf{w}_0 - \mathbf{E})$ between 0 to 1 at each iteration. The size of super-pixels in our experiments is limited to 64 pixels. But since the super-pixels are dependent on the local image structure, the number of pixels is not fixed. The number of dictionary atoms are fixed to 25% of total number of super-pixels or number of data points. Since the size of the images in one dataset is fixed, hence the number of super-pixels and dictionary size is also fixed for a dataset. In computing the similarity measure, since the sparse code histograms are normalized by the total number of super-pixels, a difference in dictionary size within a dataset, will not affect the similarity values. We use nearest neighbor classifier to obtain the class of the test images. We provide the confusion matrix for each of the three dataset and state the overall classification accuracy. In the following section we report the performance

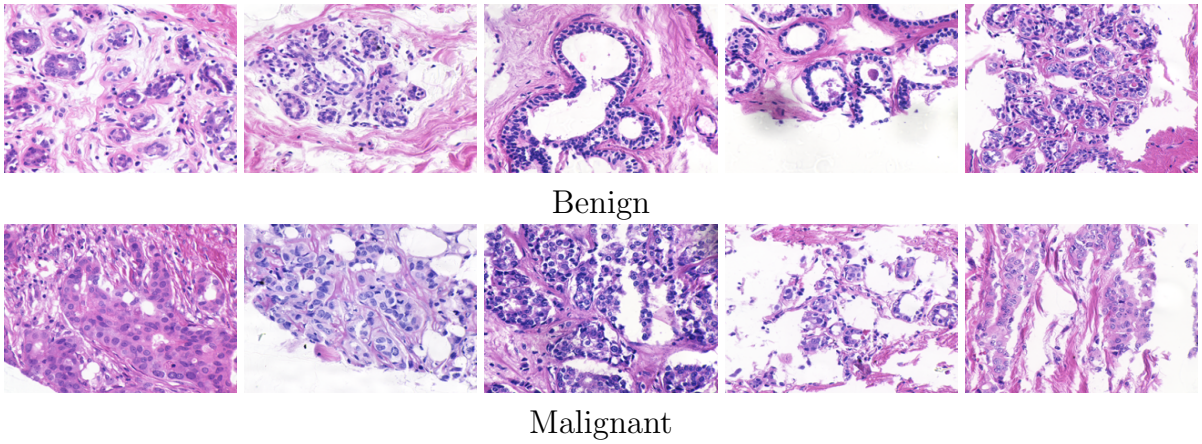
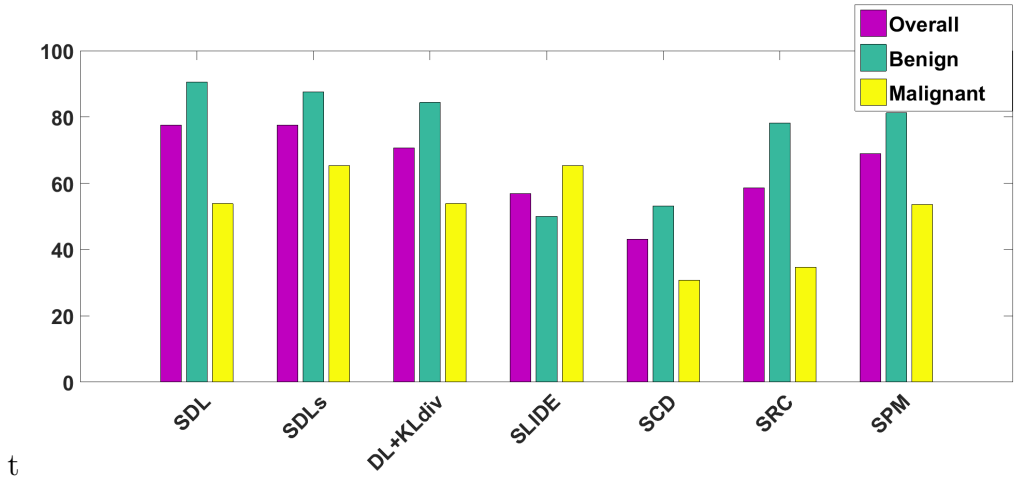


Figure 5.10: Image samples from the breast cancer tissue dataset

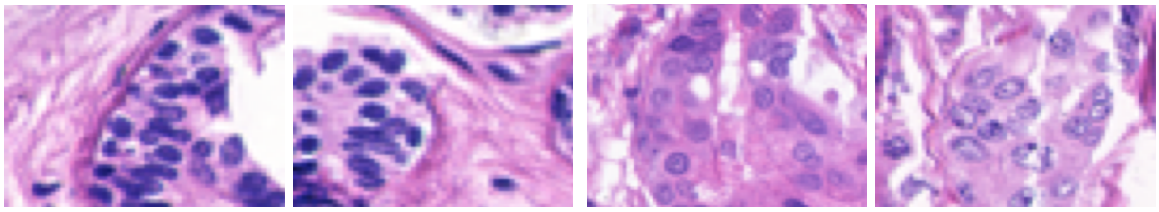


t

Figure 5.11: The bar graph shows the classification accuracy (%) obtained per class as well the overall dataset for seven different methods.

evaluation of the methods for three publicly available datasets.

ADL tissue dataset: The datset contains tissue images from three mammalian organs - lung, kidney and spleen first presented in [14]. Each of the organ images have two categories: tissue showing inflammation and normal or healthy tissue. Each category contains about 150-170 Hematoxylin and Eosin, (H&E) stained images for each of inflammation and normal tissue, of which about 25% are used for testing and the rest are used for training purpose. The images are manually annotated by pathologists from Animal Diagnostics Lab, Pennsylvania State University. More details about the dataset is available in [14]. Sample images of the dataset are shown in Fig. 5.7,5.8 and 5.9.



(a) Benign (b) Malignant

Figure 5.12: A closeup view of example breast cancer tissue images

We give the confusion matrix for the dataset in Table 5.1 for *SDL*, *SDLs* along with the comparison algorithms. The values marked in bold are the maximum accuracies achieved for the dataset. For a thorough comparison, we also provide the values of the original work SHIRC [14] and DFDL [16] for the ADL dataset. The accuracy % for these two methods are reported directly from the papers. As can seen from the table, our method outperforms the current state of the art methods. Also, *SDLs* performs better that *SDL* in majority of the categories. For tissue image classification, it is desired that diseased tissue images are classified accurately and at the same time, it is necessary that the healthy tissues are not classified as diseased. It is observed from the table, our methods perform consistently for all in classifying both healthy and diseased tissues for all the three organs.

Quantitative evaluation: The mean overall classification accuracy for ADL dataset is maximum using SDLs(95.5%) followed by SDL (94.8%), DFDL (about 94.6%, as reported in the paper), DL+KLdiv (93.8%), SPM (83.3%) and SHIRC (about 78.9%). From table 5.1, it is seen that in some scenarios, for e.g., normal kidney, inflamed lung and in spleen, DL+KLdiv, which is using KL divergence on sparse code histogram for dictionary learning, performs almost the same as either SDL or SDLs. This can be accounted for by looking at the images in Fig. 5.7,5.8 and 5.9. For these categories, the spatial distribution of the cells and contrast between the discriminative structures are not significant even to human eye, hence the saliency maps obtained are more uniform inside the image. In contrast, the tissue for a normal lung is characterized by more prominent structures and contrast created due to the large opening in alveoli [14], which can be highlighted by the saliency region detection

methods. Thus leveraging the saliency values results in an average overall increase of 13.1% and 14.3% in accuracy for the kidney dataset using SDL and SDLs respectively. For lung dataset the increase in accuracy for SDL and SDLs both show an average increase of 15.5%. For the spleen dataset, SDL shows an average increase of 15.7% and SDLs show 17.0%. For the ADL dataset, overall we achieve an average increase of 14.8% and 15.6% for SDL and SDLs respectively.

Classification accuracy versus training size: As mentioned earlier, one of the major issues in tissue image classification is the limited availability of data and inconsistent number of images per category. One of the desired properties of a classifier or the similarity measure is that it performs consistently with limited training dataset. In Fig. 5.6 (a),(b) and (c) we show the overall accuracy for the kidney, lung and spleen dataset respectively. To perform the experiment, we selected randomly 20, 40, 60, 80 and 110 images from the training set for each of the categories and plot the classification accuracy % for the different size of the training dataset. To evaluate the stability of an algorithm with changing training size, we calculate the root mean squared deviation (RMSD) given by $\sqrt{\sum_i \frac{(acc_i - acc_{i-1})^2}{n}}$, where acc_i is the classification accuracy corresponding to $i = 2 \ldots 5$ and training size 40, 60, 80 and 110. Lower the value of RMSD, more consistent is the performance of an algorithm with changing size of training dataset.

For the kidney dataset for both SDL and SDLs the accuracy increases with a RMSD of 3.4% and 3.2% respectively, while SPM shows an increase initially by then decreases with a RMSD of 3.7%. Accuracy for SLIDE on the other hand decreases with increasing number of training images with RMSD of 5.6% and SRC shows a more unstable accuracy with RMSD of 3.9%. For the lung dataset SDL shows a more steady performance with RMSD of 0.9% in comparison to SDLs, which shows an RMSD of 2.4%. SPM, SRC and SLIDE shows an RMSD of 1.6%, 2.7% and 8.7% respectively. Contrary to kidney dataset, SLIDE here shows an increase in performance with increasing training data and SPM has a more steady change in accuracy. For the spleen dataset, SDL and SDLs shows an RMSD of 0.9% and 1.1%, while

SPM, SRC and SLIDE allows an RMSD of 5.1%, 4.0% and 4.3%. In Fig. 5.6, below each figure we also provide the RMSD for individual classes and the method that shows the most steady accuracy with changing training size for each of these classes is marked in bold. It is observed that SDL shows the most steady response in majority of cases while SDLs is the second best.

For all three datasets, we notice that SDL, SDLs and SPM provide a more robust performance with different sized training. But SDL and SDLs both show an increase of in average overall accuracy of 14.8% and 15.6% for the ADL dataset over the competing methods. Thus leveraging the saliency map in dictionary learning proves to be more effective for classification.

Breast cancer tissue: The dataset contains 58 Hematoxylin and Eosin (H&E) stained images of breast cancer tissue available at *http://bioimage.ucsb.edu/research/bio-segmentation*. This dataset presented in [159] was originally obtained from Yale Tissue Microarray Facility. The dataset consists of two categories: malignant and benign with total 26 malignant and 32 benign images which are labeled by experienced pathologists. Sample images of the two classes from the dataset are shown in Fig. 5.10. A closer look at the two classes of the dataset reveals the characteristics of each of them. The benign breast cancer tissue in Fig. 5.12(a) shows that the nuclei are arranged more compactly and show textural smoothness in comparison to the malignant tissue samples in Fig. 5.12(b). In malignant tissues, the nuclei are more inhomogeneous and the staining is not as prominent as in benign tissues. Since the two types of tissues can be distinguished by the nuclei structure, detecting the salient objects can be advantageous, which would mostly highlight the nucleus regions as is evident from Fig. 5.3(a), (b) and (c).

For classification of the dataset, we perform leave one out method, i.e., we select one image as a test image and compare with all other images in the training set and classify based on 1-nearest neighbor classifier using the similarity measure defined in eq. (5.18). The confusion table is given in table 5.2. While SDLs performs better in detecting malignant

Table 5.2: Confusion matrix % for breast cancer tissue dataset

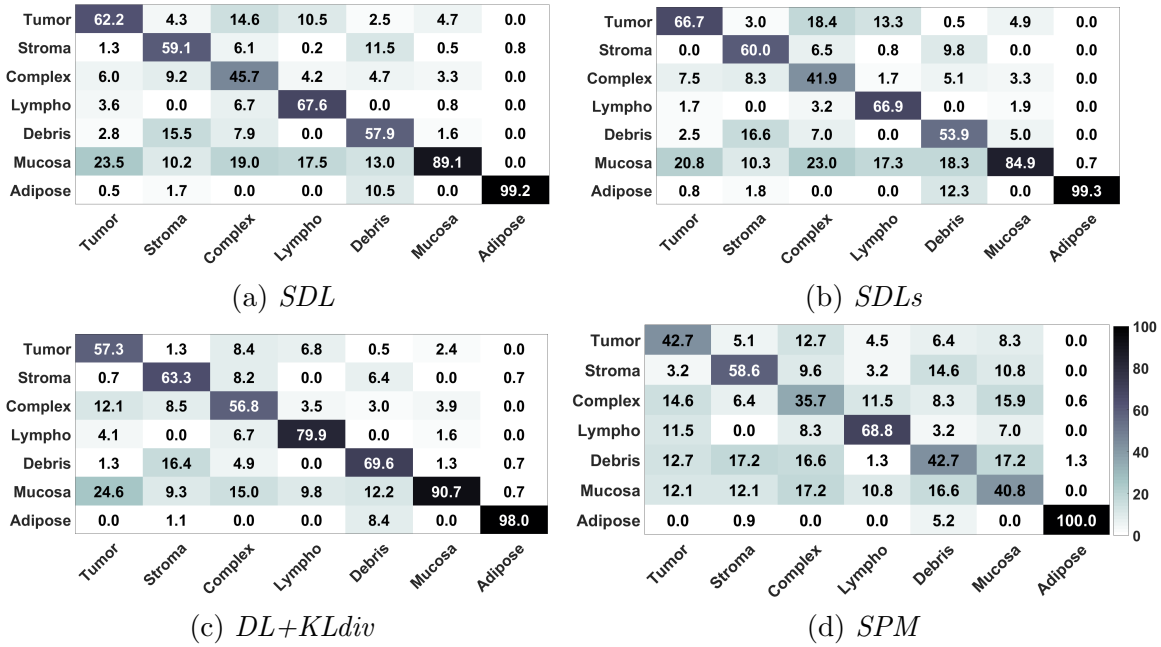
	Method	Benign	Malignant
Benign	*SDL*	**90.6**	9.37
	SDLs	87.5	12.5
	DL+SC KLdiv	84.34	15.66
	SLIDE [10]	50	50
	SCD [11]	53.13	46.87
	SRC** [15]	78.13	21.87
	SPM [139]	81.25	18.75
Malignant	*SDL*	38.46	61.5
	SDLs	34.62	**65.4**
	DL+SC KLdiv	46.15	53.85
	SLIDE [10]	36.62	63.38
	SCD [11]	69.23	30.77
	SRC** [15]	65.38	34.62
	SPM [139]	46.42	53.58

**Experiments with full images.

tissues, SDL classifies benign tissues with greater accuracy. The individual class recall and overall classification accuracy is given in Fig. 5.11. For both SDL and SDLs, the classification accuracy for the breast cancer tissue dataset obtained is 77.6% while that using DL+kldiv is 70.7% and SPM is 69%. For this dataset SDL and SDLs achieve and average overall increase in accuracy by 17.92% for SDL and 17.93% for SDLs.

From table 5.2 we notice that the performance for the benign tissues are always greater than that of malignant tissue. One of the reasons is possible due to the fact that all the nuclei in the tissues annotated as malignant show malignancy as well as benign properties, thereby increasing the chances of mis-classification. One of the ways to tackle this problem is to detect the nuclei and classify them individually and finally classify based on the percentage of nuclei in the tissue showing malignancy properties. This cell level classification was performed in one of the experiments in [159] and shows promising results. However our algorithm is limited in this context and can only perform classification at an image level.

Colon cancer tissue: For the previous two datasets, we show a binary classification problem. For both ADL and the breast cancer tissue dataset, the tissue images belong to



(a) *SDL* (b) *SDLs*

(c) *DL+KLdiv* (d) *SPM*

Figure 5.13: The confusion matrix (%) for the colorectal tissue dataset is shown in the figure for the four methods *SDL*, *SDLs*, DL+KLdiv, SPM [139].The predicted class is given along the y-axis.

two classes: normal and diseased. For the colon cancer tissue classification, we extend our application to multi-class classification problem. The colon cancer tissue dataset obtained from [160] contain eight different types of tissue obtained from human colorectal cancer sites. As mentioned in [157], the samples obtained from human tumors are complex structures and consists of various tissue types. The progression of the disease can be evaluated by analyzing the tissue composition in tumors. Manually quantifying the tissue composition in the tumor images is time consuming and expensive, which necessitates automated analysis of the images. The different tissue types can be distinguished by the patterns they exhibit in the H&E stained slides on the tumor samples. As demonstrated in the paper, textural analysis of the histo-pathological images prove to be often effective in distinguishing the tissue types.

The dataset contains 8 different tissue categories obtained from H&E stained slides of colorectal tumor samples. The tissue categories was created by manually annotating smaller overlapping regions of size 150×150 from these samples. The eight categories include: tumor tissues, stroma, complex structured stroma, lympho, debris, mucosa, adipose and background.

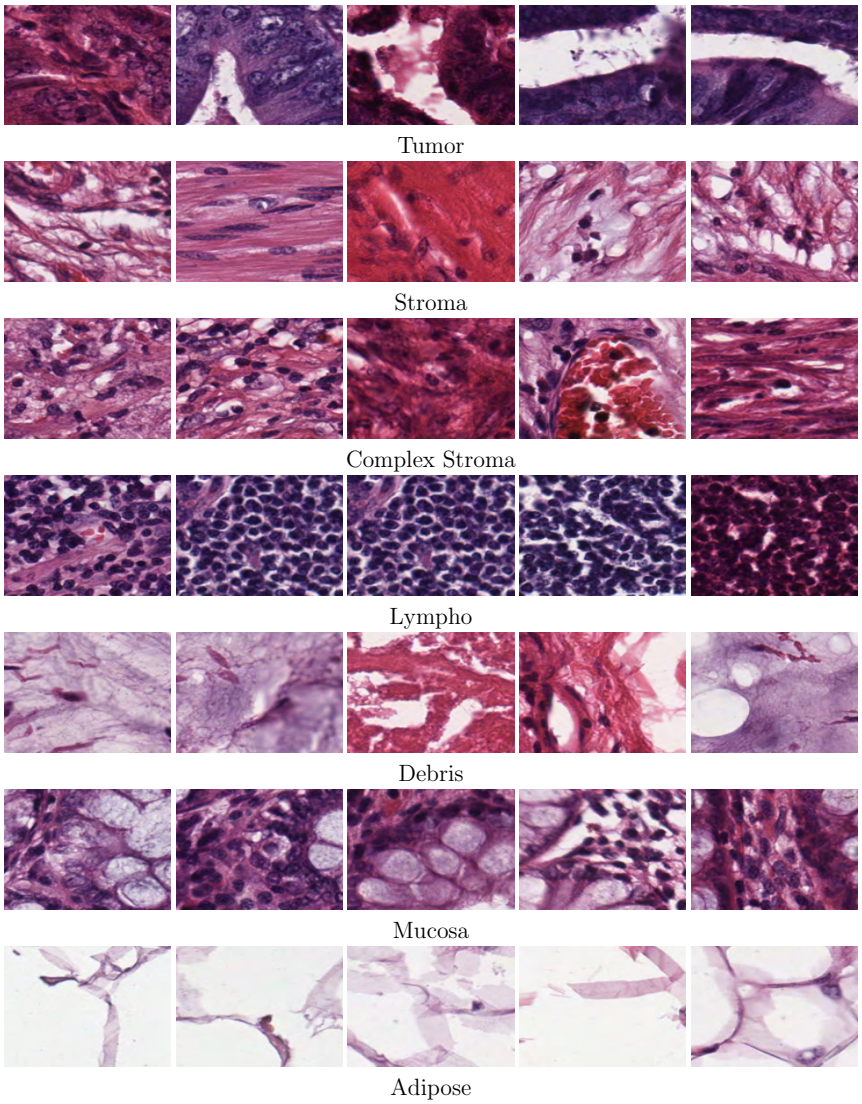


Figure 5.14: (a) Sample images from the colon cancer tissue dataset.

Sample images from 7 classes are shown in Fig. 5.14. For our experiments, we classify the tissues in these seven classes, since the last category or the background do not contain any objects, or do not show any patterns which makes it impractical to use saliency on such images. There are total 4375 images for the seven classes, with 625 images per class. For our experiments, we use 468 images per class as training images and the rest 157 as test images.

The confusion matrix for the dataset is given in table 5.13. For testing, we randomly select a subset of 450 training samples per class and 150 images from the test set. We perform 10-fold experiments, each time selecting a different combination of training and

Table 5.3: Retrieval accuracy % for military vehicle dataset 1

	Category 1	Category 2	Category 3	Category 4
SDL	41.3	**92.4**	61.1	38.2
SDLs	47.8	88.5	**77.8**	41.2
DL+KLdiv	50	91.0	58.3	52.9
SLIDE	54.4	83.4	41.8	**66.2**
SLIDEs	47.8	65.4	47.2	26.4
SLIDEdl	36.3	76.9	28.1	7.6
SPM [139]	47.8	87.1	44.4	50.0
ID [144]	**71.4**	16.7	16.1	4.3

Category 1 Category 2 Category 3 Category 4

Figure 5.15: Sample images from dataset 2 from each of 4 categories

test images per class and performing experiments using 20 nearest neighbor classifier. The columns correspond to test class and the rows represent predicted category. Since more competing performance was obtained using DL+KLdiv and SPM for previous two datasets, hence we compare the classification accuracy for this dataset using these two methods. As we can see from the confusion matrix, *SDL* and *SDLs* has a higher recall in classifying tumor tissues. *SDL* and *SDLs* shows an average increase of 7% and 11% in recall respectively for tumor tissue in comparison with other methods. On the other hand, the precision on average decreased by 0.1% for both *SDL* and *SDLs* while the F-measure increased by 4.0% and 5.7% respectively.

5.3.4 Application to military image classification

Experiments for image retrieval using the saliency guided dictionary learning framework for image similarity evaluation (*SLIDE*) were performed for two datasets.

Dataset 1: It contains 190 images of military vehicles and weapons. The images are divided into four categories - fighter jets, light and heavy duty land vehicles, naval vessels and

Table 5.4: Retrieval accuracy % for military dataset 2

	Category 1	Category 2	Category 3	Category 4
SDL	41.1	33.2	**52**	**74.4**
SDLs	45.8	45.8	48	70
DL+KLdiv	**58.3**	16.7	**52**	72
SLIDE	50.4	45.8	32.5	66.2
SLIDEs	50	20.8	20.2	34.0
SLIDEdl	32.6	**76.9**	27.8	7.6
SPM [139]	41.6	33.3	20	68.8
ID [144]	58.3	12.5	48.0	34.4

Category 1 Category 2 Category 3 Category 4

Figure 5.16: Sample images from dataset 3 from each of 4 categories

weapons with 46, 78, 36 and 30 images, respectively. The images are obtained from various sites through the Google search engine. The major challenge with this dataset lies in the following aspects. First the variability in background- the objects are imaged with different backgrounds. Second there may be multiple objects in one image. In both these scenarios, saliency detection aids the process of emphasizing the relevant regions while computing the similarity. The third is the variability in the object - the objects vary in shape, orientation and texture rendered due to camouflage patterns. To accommodate diversity, a combination of color and texture features are used. The sample images of this dataset are shown in Fig. 5.15 and the retrieval results in Table 5.3.

Dataset 2 contains 120 images of different military vehicles divided into four categories of fighter jets and helicopters, light utility land vehicles, naval vessels and heavy duty land vehicles with 24, 24, 25 and 47 images respectively. It contains images of land vehicles in two different categories, which makes the dataset more challenging. The retrieval results of for this dataset are given in Table 5.4 and sample images are shown in Fig. 5.16.

For all the datasets, a combination of color and Gabor features [24] are used as described

in Section 5.3.1. Approximately $N_i = 500$ superpixels were extracted from each image and the dictionary size was fixed to $K = 100$. The results are evaluated using a 'leave one out' strategy. We compare the retrieval accuracy % ($\frac{\#\ images\ with\ correct\ prediction}{\#\ images\ in\ that\ category}$) of *SLIDE* with the similarity measures proposed in [11, 144]. It can be seen that *SLIDE* outperforms the other algorithms in the majority of cases (77.8%). We also observe that our similarity measure performs significantly even without incorporating salient object detection.

5.4 Discussion

In this paper we introduce a novel method of saliency based sparse coding and dictionary learning for computing image similarity between a pair of images. The method leverages salient object detection technique to obtain prominent features form images. The designed dictionary learning technique exploits these saliency to obtain the sparse codes and set of basis function for image representation such that the more salient image regions has more contribution in the dictionary. We show two ways to learn the saliency and dictionary learning in this paper, the first uses a constant saliency map to update the dictionary, the second method updates the saliency up with the dictionary and sparse codes by incorporating a spatial smoothness constraint.

The similarity measure is designed based on the fact that a dictionary learned from an image will have similar contribution in representing another image from the same category. For the test phase in classification, the saliency weighted dictionary and sparse codes are learned for the query image and then for each pair of query and training image the features are represented with each other's dictionary. The final four sets of *self* and *relative* sparse codes are compared using KL divergence. The test image is classified based on the image that gives minimum similarity score.

The methods were applied in the application of histopathological tissue image classification. We perform experiments on three different datasets. The first dataset (ADL) contains diseased

and normal tissues of three mammalian organs - kidney, lung and spleen obtained from [14,16]. The second dataset contains malignant and normal breast cancer tissue dataset [159] and the third contains colon cancer tissues [160]. For all the three datasets, we show that including the saliency improves the classification accuracy and also performs significantly better than the state of art methods. Additionally, for the military vehicle detection dataset, our methods outperforms the stae of the art.

SDL and SDLs both can also be further extended in as a supervised method, which can be advantageous for multi-class classification problem. The feature saliency of a local image region can be computed based on information of all the local image features in one category and then learn one dictionary for each category. When computing saliency of individual images, it does to provide any information about co-saliency of regions between two images and hence similar regions in two images can have different saliency values. The supervised approach, where the saliency can be normalized of a particular category. We have shown applications of tissue image classification, but the methods are not limited to histo-pathological image analysis. Since the dictionary is learned in an unsupervised manner, the similarity measure designed can be applied in other applications e.g., clustering event detection.

Chapter 6

Event detection by leveraging region saliency

The previous two chapters were focused on spatial feature selection from images, which are used to enhance more discriminative features based on the spatial relationships with neighboring regions. In this chapter, we will extend this method to a 3D framework to analyze videos for unusual event detection. In contrast to dealing with single images, both spatial and temporal feature relation need to be considered. This problem can be tackled in two different ways. First, a frame-by-frame analysis can be performed. Second, the entire video can be treated 3D data, and a volumetric analysis can be attempted. The spatio-temporal analysis of videos can be exploited in different applications e.g., object motion tracking, activity recognition, event detection. This chapter emphasizes unusual and hazardous event detection from video.

Event detection generally denotes any change in the temporal sequence that does not adhere to the original pattern of the sequence. This process can indicate unusual or suspicious behavior of certain objects, malicious activity, change in usual trajectory patterns, accidents, *etc.* A sub problem in this category is adverse event detection that refer to unpredictable incidents or events that occur in a video abruptly and can cause damage to life and property.

These types of events do not follow any regular pattern and their occurrence in a video sequence is occasional for e.g., road accidents, sudden fire, *etc.* In a surveillance system, automatically detecting such rare and adverse incidents can help in further analysis and examination of the situation. Additionally, this detection would help in delivering timely aid and relief.

Traditional approaches for change or event detection generally rely on spatio-temporal feature [94,161], saliency analysis [104–106], motion trajectory pattern detection [100,162], or background subtraction methods [102,163,164]. A more recent trend in event detection is based on sparse and low rank representation approaches [1,2,110–115,165]. The aforementioned methodologies are developed based on the assumption that the temporal change in background is limited. As a consequence, the background can be analyzed by a low rank matrix.

Recently, to obtain a more sophisticated supervision in safety and security, car-mounted and wearable cameras have been introduced. However, in scenarios where the videos are captured using wearable/hand-held or camera fitted on a car, camera jitter, camera motion add to changing dynamic background. In such scenarios, the background becomes dynamic hence in addition to dealing with detecting an event in the video, one need to take care of occlusion, camera motion and highly changing background. In such scenarios, the background becomes dynamic and the assumption of slow changing background is then invalid. In order to increase robustness to the background changes we propose a spatio-temporal analysis approach that combines a saliency-driven sparse representation technique. The spatio-temporal saliency aids identifying distinctive features which can act as an indicator of significant spatio-temporal changes. The saliency driven sparse representation assist in discriminative between events and changes occurring due to background motion or occlusion.

Objective

To address the abnormal event detection problem while accounting for significantly dynamic background we first propose a frame-by-frame analysis. In order to increase robustness to the background changes we propose a spatio-temporal analysis approach (SSPARED). The method combines a saliency-driven sparse representation of each frame with an information theoretic metric, based on the Kullback-Leibler divergence. Similar to our image classification framework, we employ a cross-dictionary representation between frames and the temporal changes between consecutive sparse representations is quantified by the K-L divergence. We envision that the frames demonstrating significant change in the representation would indicate the occurrence of events.

While the consecutive frame analysis aids in solving the problem, the method can be computationally expensive. Moreover, it does not account for the temporal change in the saliency map. In order to deal with this, we perform a volumetric analysis of the video by local sparse and low rank representation of the frames aided by spatio-temporal saliency. Spatial saliency detection [126, 166] generates a map mimicking human visual attention model that highlights unique regions. Spatio-temporal saliency detects regions in video volume which are distinct from other locations, spatially as well as temporally. The saliency detection captures any changes in scene which occurs with time but not necessarily capture the abnormal events exclusively, especially when changes can occur due to background motion. We exploit this change detection capability of spatio-temporal saliency by integrating it with a sparse and low rank representation in the proposed method SpLoRed. The method is based on the idea that depending on the region saliency, a local region can be considered as a part of an event or background. Higher saliency regions are more probable of being a part of an event and is modeled as sparse linear combination of some basis function learned from the data. Regions with low saliency values can be then considered as part of background and are modeled as a low rank matrix. Thus, the saliency balances the event and background. The final event detection is performed by analyzing the sparse representations and the background

approximation.

6.1 SSPARED:Saliency and sparse code analysis for rare event detection in video

As stated earlier, this method solves the detection problem by analyzing consecutive frames in a video. In this method, we exploit the saliency guided dictionary learning technique to obtain the sparse code for the consecutive frames. The saliency based dictionary learning is solved in the similar manner as described in the method *SDL* discussed in section 5.1.1 in the previous chapter. For a pair of consecutive frames, the local features are represented with respect to each other's dictionary and the sparse code histograms are obtained. Temporal change in the KL-divergence of the sparse code histograms are then analyzed to quantify the change in the scenes and detect the occurrence of the event.

For consecutive frames at t and $t-1$, the features extracted are denoted by $\mathbf{Y}_t$ and $\mathbf{Y}_{t-1}$. The dictionaries learned for frames t and $t-1$ are $\mathbf{D}_t$ and $\mathbf{D}_{t-1}$ respectively and their corresponding sparse representations are $\mathbf{X}_t$ and $\mathbf{X}_{t-1}$. We call them *self sparse codes*. When $\mathbf{Y}_t$ and $\mathbf{Y}_{t-1}$ are represented with respect to $\mathbf{D}_{t-1}$ and $\mathbf{D}_t$, the sparse representation $\mathbf{X}_{t|t-1}$ and $\mathbf{X}_{t-1|t}$ are called *relative sparse codes*. $\mathbf{W}_t$ and $\mathbf{W}_{t-1}$ are the saliency matrices whose diagonal contain the saliency values for the corresponding superpixels. In the next sections, we describe how we compute and analyze the temporal sparse code histograms. Finally, we provide results of our method as well as comparison to other methods, using four different videos captured under different conditions and showing various hazardous events.

6.1.1 Rare event detection with sparse code histograms

To detect the time of occurrence of unusual event we perform a comparison between each two consecutive frames by representing each frame using the dictionary learned from the other. The histograms formed from the sparse representation are then compared using the

Algorithm 4 Algorithm for rare event detection

For each frame $t \geq 0$

1. Superpixel segmentation to obtain $\mathbf{Y}_t = [\mathbf{y}_1....\mathbf{y}_N]$
2. For each superpixel i compute saliency using 5.2 and obtain $\mathbf{W}_t$. N denotes the total number for superpixels in one frame.
3. Obtain the dictionary $\mathbf{D}_t$ and the corresponding sparse codes $\mathbf{X}_t$ using the superpixel features $\mathbf{Y}_t$ and the saliency $\mathbf{W}_t$
   a. The dictionary $\mathbf{D}_t$ is initialized by choosing K most salient features.
   b. The sparse codes X_t are obtained by solving (5.3) keeping $\mathbf{D}_t$ fixed.
   c. The dictionary $\mathbf{D}_t$ is updated keeping $\mathbf{X}_t$ fixed, by solving (5.4) which for each frame is written as,$\sum_{i=1}^{N}(w_i)_t\|(\mathbf{y}_i)_t - \mathbf{D}_t(\mathbf{x}_i)_t\|_2^2$

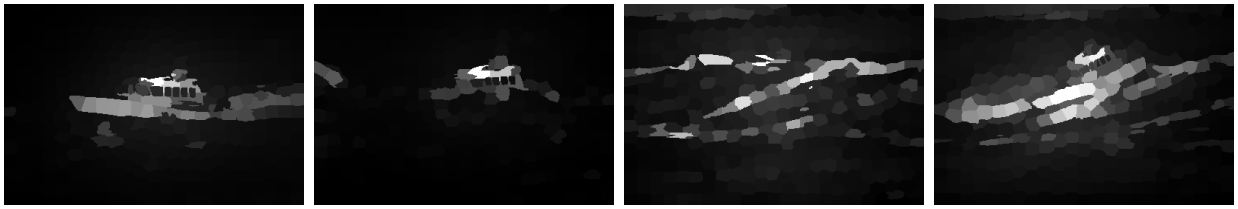
$$
\begin{aligned}
&= \sum_{i=1}^{N} \|\sqrt{(w_i)_t}(\mathbf{y}_i)_t - \sum_{j=i\neq k}^{N} \sqrt{(w_i)_t}(\mathbf{d}_j)_t(\mathbf{x}_i^j)_t - \sqrt{(w_k)_t}(\mathbf{d}_k)_t(x_i^k)t\|_2^2 \\
&= \sum_{i=1}^{N} \|(e_k)_i - (w_k)_t\mathbf{d}_k x_i^k\|_2^2 \;=\; \|(E_k)_t - \sqrt{(w_k)_t}(\mathbf{d}_k)_t\mathbf{X}_t^k\|_F^2 \qquad (6.1)
\end{aligned}
$$

   $\mathbf{d}_k$ is obtained by taking the singular value decomposition of $(E_k)_t$=$\mathbf{U\Sigma V}$ and $(\mathbf{d}_k)_t = \mathbf{U}(:,1)$.

For $t > 0$

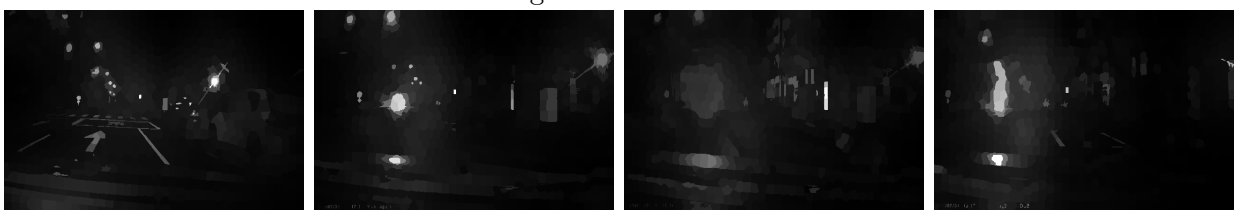
1. Obtain $\mathbf{X}_{t|t-1}$, $\mathbf{X}_{t-1|t}$ as the representation fo $\mathbf{Y}_t$ and $\mathbf{Y}_{t-1}$ with respect to $\mathbf{D}_{t-1}$ and $\mathbf{D}_t$ respectively.
2. Compute histograms H_{I_t}, $H_{I_{t|t-1}}$, $H_{I_{t-1}}$,$H_{I_{t-1|t}}$ using (6.2).
3. Compare the histograms by KL-divergence method using 6.4
4. Compute the change in the KL-divergence, which if greater than a threshold τ indicates an unusual event has occured in the scene.

a. Original frames video 1



b. Saliency map video 1

a. Original frames video 2



b. Saliency map for video 2

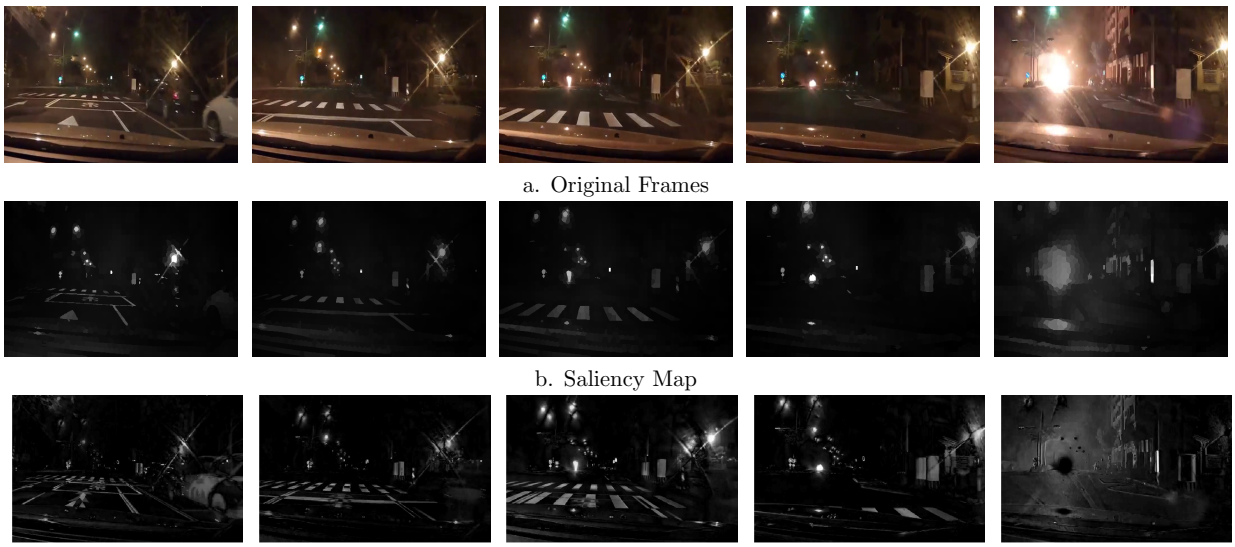
Figure 6.1: The figure shows original video frames in (a) and (c) for video 1 and 2 respectively. The corresponding saliency maps for a video sequence 1 and 2 are shown in (b) and (d) respectively.

Kullback–Leibler (KL) divergence method. Applying (5.3) to the images at frames t and $t-1$, we can learn, for the features $\mathbf{Y}_t$ and $\mathbf{Y}_{t-1}$, the *self sparse codes* $\mathbf{X}_t$ and $\mathbf{X}_{t-1}$ by solving the following,

$$\min_{(\mathbf{x}_i)_t} \|(\mathbf{x}_i)_t\|_0 \quad \text{s.t.} \quad \|(\mathbf{Y}_t - \mathbf{D}_t X_t)\mathbf{W}_t^{\frac{1}{2}}\|_2^2 \leq \epsilon$$

$$\min_{(\mathbf{x}_i)_{t-1}} \|(\mathbf{x}_i)_{t-1}\|_0 \quad \text{s.t.} \quad \|(\mathbf{Y}_{t-1} - \mathbf{D}_{t-1} X_{t-1})\mathbf{W}_{t-1}^{\frac{1}{2}}\|_2^2 \leq \epsilon$$

When $\mathbf{Y}_t$ and $\mathbf{Y}_{t-1}$ are represented with respect to $\mathbf{D}_{t-1}$ and $\mathbf{D}_t$, the *relative sparse codes* $\mathbf{X}_{t|t-1}$ and $\mathbf{X}_{t-1|t}$, are obtained by solving the following, $\forall i \in 1, 2, \ldots, N$



a. Original Frames

b. Saliency Map

c. Background Subtraction using median of previous frames

Figure 6.2: Sample frames for video 2 (a) with a comparison between the saliency maps (b), the median background subtraction approach in (c).

a. Original Frames

b. Saliency Map

c. Background Subtraction

Figure 6.3: Sample frames for 4^{th} video (a) with a comparison between the saliency maps (b) and the median background subtraction (c) approaches.

$$\min_{(\mathbf{x}_i)_{t|t-1}} \|(\mathbf{x}_i)_{t|t-1}\|_0 \text{ s.t.} \|(\mathbf{Y}_t - \mathbf{D}_{t-1}X_{t|t-1})\mathbf{W}_t^{\frac{1}{2}}\|_2^2 \leq \epsilon$$

$$\min_{(\mathbf{x}_i)_{t-1|t}} \|(\mathbf{x}_i)_{t-1|t}\|_0 \text{ s.t.} \|(\mathbf{Y}_{t-1} - \mathbf{D}_{t}X_{t-1|t})\mathbf{W}_{t-1}^{\frac{1}{2}}\|_2^2 \leq \epsilon$$

Temporal sparse code histogram

As shown in [11, 154], compressibility of the sparse representation can be exploited in computing similarity between images. After representing the frame using the histogram computed from the sparse codes x_i, we exploit this compressibility while taking into account the contribution of each dictionary elements. The advantage of this representation is that the histograms can be further used as a representative feature of the images. If I_t is the frame at time t, the *self sparse code histogram* is computed as

$$H_{I_t}(k) = \|\mathbf{X}_t^k\|_0 \tag{6.2}$$

where $\mathbf{X}_t^k$ is the k^{th} row of the matrix $\mathbf{X}_t$ that contains the sparse codes representing I_t. The dictionary atoms act as the bin centers of the histogram and the number of features that share the dictionary atom constitute the bin frequency. The histogram is normalized by the total number of superpixel features.

The histograms constructed from the *relative sparse codes* can be obtained by

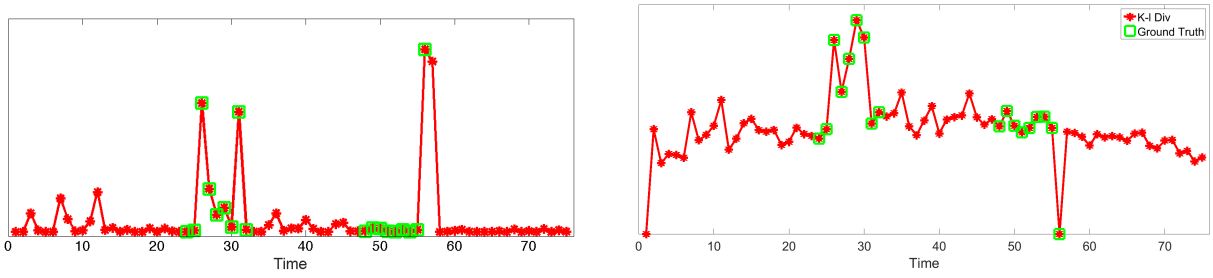
$$H_{I_{t|t-1}}(k) = \|\mathbf{X}_{t|t-1}^k\|_0; \;\; H_{I_{t-1|t}}(k) = \|\mathbf{X}_{t-1|t}^k\|_0$$

We call $H_{I_{t|t-1}}$ and $H_{I_{t-1|t}}$ *relative sparse code histograms*

Change detection using sparse code histogram

When the change between two consecutive frames is significant, the change in the actual sparse code histograms H_{I_t}, $H_{I_{t-1}}$ and the *relative sparse code histograms* $H_{I_{t|t-1}}$, $H_{I_{t-1|t}}$ will also be significant. We can quantify this change by using the KL divergence [138] to measure the difference between the histograms:

$$KL(p||q) = \sum_i p(i) \log \frac{p(i)}{q(i)} \tag{6.3}$$



(a) (b)

Figure 6.4: (a) shows the KL divergence $\mathcal{D}(I_t, I_{t-1})$ between frames of video 1. The ground truth of the actual event is marked in green. In (b) the change in the KL divergence $\eta(t)$ for video 1. The ground truth of the actual event is marked in green.

In particular, we compute the final KL divergence as an aggregate of the KL divergence of histogram pairs

$$\mathcal{D}(I_t, I_{t-1}) = KL(H_{I_t}||H_{I_{t-1|t}}) + KL(H_{I_{t-1|t}}||H_{I_t}) + KL(H_{I_{t-1}}||H_{I_{t|t-1}}) + KL(H_{I_{t|t-1}}||H_{I_{t-1}})$$

The difference $\eta(t) = \mathcal{D}(I_t, I_{t-1}) - \mathcal{D}(I_{t-1}, I_{t-2})$ is computed and used to identify significant changes occurring between frames $t-1$ and t. Assuming that no event occurs within the first n frames, τ is computed as maximum of $\eta(t)$ for $t = 1...n$. For the following frames, if $\eta(t) > \tau$, t is detected as the time of occurrence of an unusual event. In case of events occurring within the first n frames, the approach still works and will detect the end of the event. In Fig.6.4(a) and Fig.6.4(b) we show respectively $\mathcal{D}(I_t, I_{t-1})$ and $\eta(t)$ plotted for video 1 (see Section 6.2). The ground truth for the event occurrence is shown in green. As it can be seen from the plot, the event causes considerable alteration in scene and is correctly identified by our approach. The algorithmic description of the method is given in algorithm:4.

6.2 Experimental Results

Experiments were performed on four video sequences demonstrating varying types of rare events under different scenarios. We compare our method with ADM [1] and DRMF [2], two

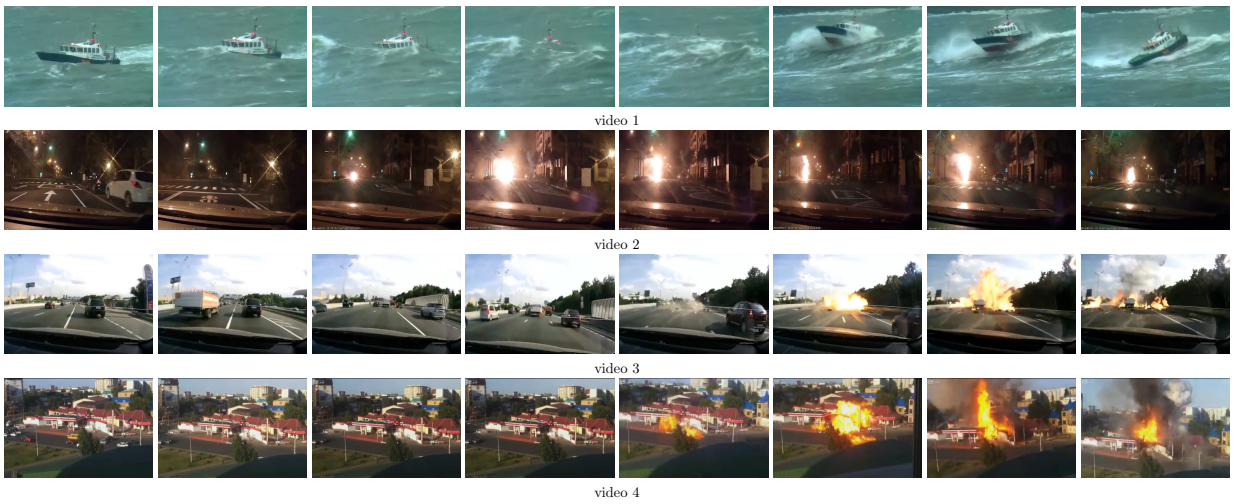


Figure 6.5: Sample videos showing examples of rare events.

methods based on the background subtraction approach. The change in the scene is detected by evaluating, the absolute error (E_t) between the observation and the estimated background, for each time frame t. The temporal change of E_t, $E_t - E_{t-1}$ is used to detect the rare event in the sequence.

In video 1 (76 frames), the two events are the disappearance and reappearance of the object. In this case, the change in the scene does not occur abruptly in one frame, but gradually over 18 frames and the ground truth consists of the range of consecutive frames from the start of the occlusion event to its end. The detection results for this sequence are presented in Fig. 6.6. The ground truth is in green, the events detected by SSPARED are marked in red, and those by ADM and DRMF are denoted by black and blue respectively. As it can be seen from the figure, SSPARED demonstrates significantly better results by accurately detecting both events whereas ADM and DRMF fail to detect the first event (disappearance) and produce a false alarm rates ($\frac{\#frames\ false\ event\ detected}{\#frames\ with\ no\ event}$) of 5% and 17% respectively.

Video 2 in Fig. 6.7 is captured by a camera fitted to a car, and the event is marked by a blast occurring in a gas line. The video sequence shows significant background variation caused by the motion of passing vehicles and the changing view along the road. In this video,

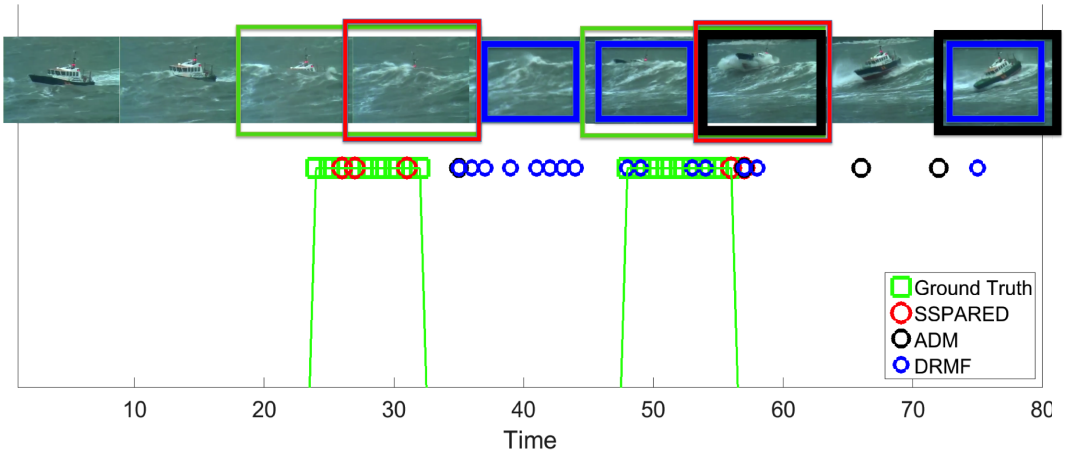


Figure 6.6: Detection results for video 1 using SSPAPRED (red), ADM [1] (black) and DRMF [2] (blue). The ground truth is shown in green. Some sample frames of the videos sequences are provided along with the detection.

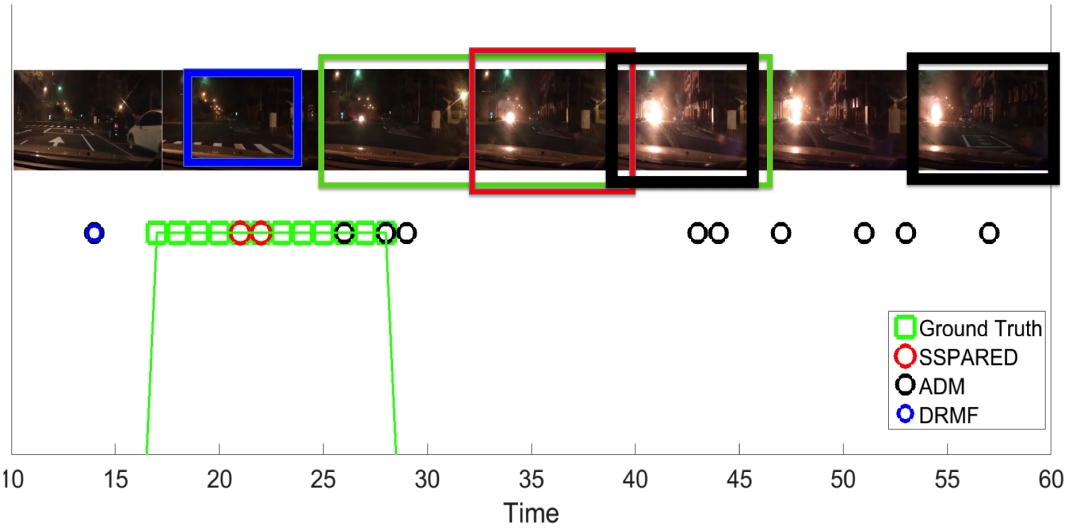


Figure 6.7: Detection results for video 2 using SSPAPRED (red), ADM [1] (black) and DRMF [2] (blue). The ground truth is shown in green. Some sample frames of the videos sequences are provided along with the detection.

12 frames are identified as ground truth, and 48 frames contain no event. It could be argued that the fire event continues after the frames selected as ground truth, but, in practice, we may be interested in detecting the initial occurrence rather than the actual temporal extent of the event. Hence, the ground truth for the event is marked as the first few frames of its occurrence instead of the entire temporal stretch of the event. In this sequence SSPARED detects the occurrence of incident accurately and yield 0% false alarm rate. ADM detects the

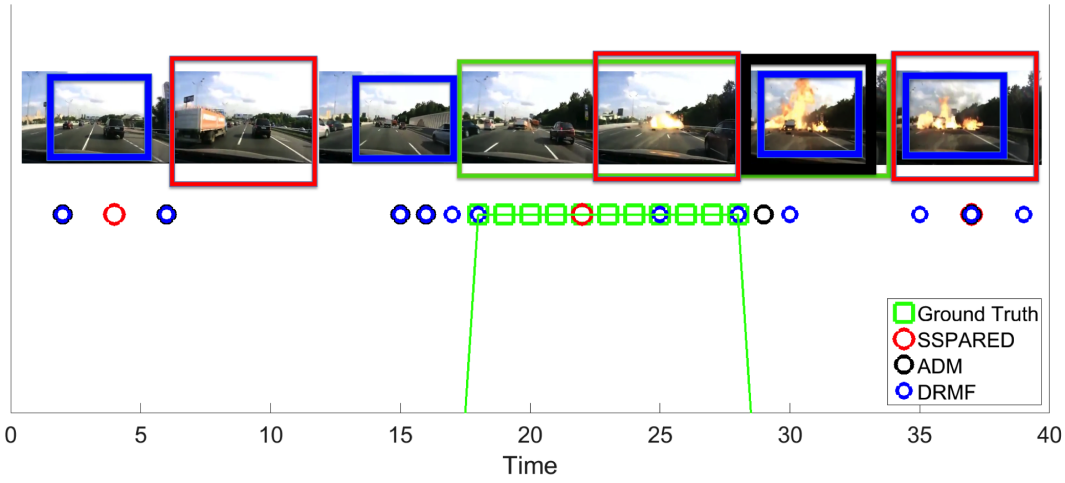


Figure 6.8: Detection results for video 3 using SSPAPRED (red), ADM [1] (black) and DRMF [2] (blue). The ground truth is shown in green. Some sample frames of the videos sequences are provided along with the detection.

event but yields a false alarm rate of 16%. DRMF fails to detect the event and yields a false alarm rate of 2%.

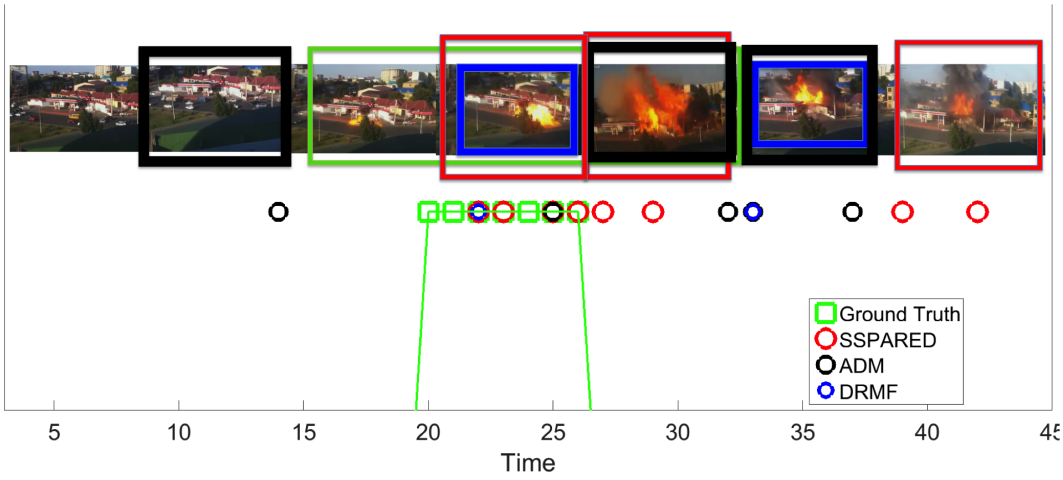


Figure 6.9: Detection results for video 4 using SSPAPRED (red), ADM [1] (black) and DRMF [2] (blue). The ground truth is shown in green. Some sample frames of the videos sequences are provided along with the detection.

Video 3 in Fig. 6.8 is obtained by a camera fitted to a car as well, but the video is captured during daytime, thus creating even larger variation in the background. The video has 11 frames marked as ground truth, and the remaining 28 frames contain no event. The event involves a vehicle accident followed by a fire. While SSPARED detects the event accurately,

Table 6.1: Confusion matrix (%) for event detection in video

		Event	No Event
Event	SSPARED [21]	**100**	0
	ADM [1]	60	40
	DRMF [2]	60	40
No Event	SSPARED [21]	5.8	**94.2**
	ADM [1]	15.8	84.2
	DRMF [2]	13.9	86.1

it also creates a false alarm rate of 7%. ADM fails to detect the event and creates false alarm rate of 12%. DRMF detects the event accurately but creates false alarm rate of 32%. The main cause for the high false alarm rate can be found in the appearance of other vehicles within the scene.

Video 4 in Fig. 6.9(d) presents the unusual event of a fire explosion in a gas station captured by a hand-held camera. In this case, the jitter caused by the hand motion is the main reason for the changes in the background. The sequence has 7 frames marked as ground truth and 38 frames with no event. SSPARED detects the event but produces false alarm rate of 10% . Interestingly, the false alarms are caused mainly by the changing appearance of the fire. ADM and DRMF detects the event but generate false alarm rate of 10% and 2%.

The event detection results for the four videos using SSPARED and comparison algorithms are consolidated and presented as a confusion matrix in 6.1.

Discussion

In SSPARED we propose a novel framework for rare event detection that leverages a dictionary learning approach where features are weighted using saliency maps. The proposed method has the advantage of localizing the compact representation towards the presence of salient features hence highlighting occurring changes within video sequences independently of the less relevant changes in the background. By exploiting an information theoretic approach, we have shown that the histograms of the sparse codes can be used to precisely detect the

time of occurrence of an event in a video achieving improved performances with respect to existing state-of-the-art methods.

SSPARED requires a frame by frame analysis in which a dictionary is learned for each frame, and also the sparse codes are computed by cross-dictionary representation of consecutive frames. This frame-by-frame makes the method computationally expensive. Hence, we propose a method which analyzes the video as a volume to determine the temporal occurrence of an event. In the following section, we discuss the sparse and low rank based volumetric analysis of video with application to rare event detection.

6.3 SpLoRed:Spatio-temporal saliency guided sparse and low rank representation

As stated earlier, sequence of video frames can be represented as a combination of a low rank matrix and a sparse matrix. The low rank matrix provide an estimation of the background while the sparse matrix account for the changes occurring in the scene. These methods have been effective in detecting scene changes or events in surveillance videos [1, 2, 110–115, 165] with static background. In our application due to presence of jitter, motion of the background, approximating the background a low-rank matrix can be erroneous. As shown in Fig. 6.10(a) and 6.11(a), from the temporal change of the frames, it is difficult to say at what time the event is occurring as well as if the change is due to an event of changes in background.

In general, it can be seen that when analyzing the temporal changes in local regions of a video, some regions demonstrate limited variation in scene while others show significant changes as shown in Fig. 6.10(b) and 6.11(b). For e.g., in Fig. 6.10(b), the events are most prominent in the blocks 2 and 7, while significant temporal change is demonstrated by blocks 2, 4, 5 and 7. A similar pattern is noticed in Fig. 6.11. This implies that the temporal changes in the blocks may result from a changing background, an actual event or a combination of both. Additionally, when an event occurs in the video, it can either occur in a block partially

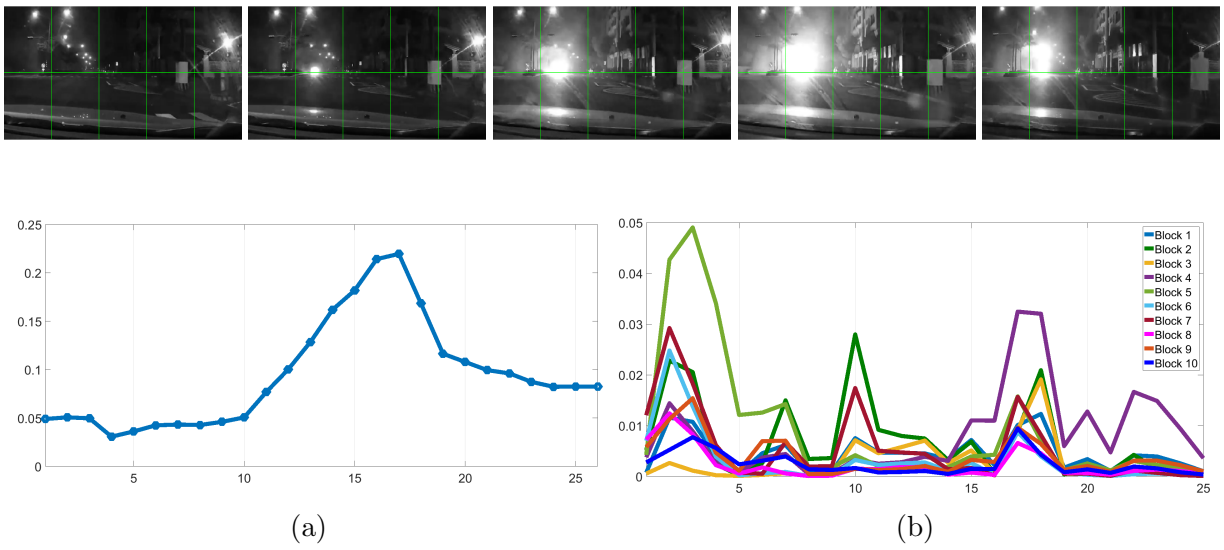
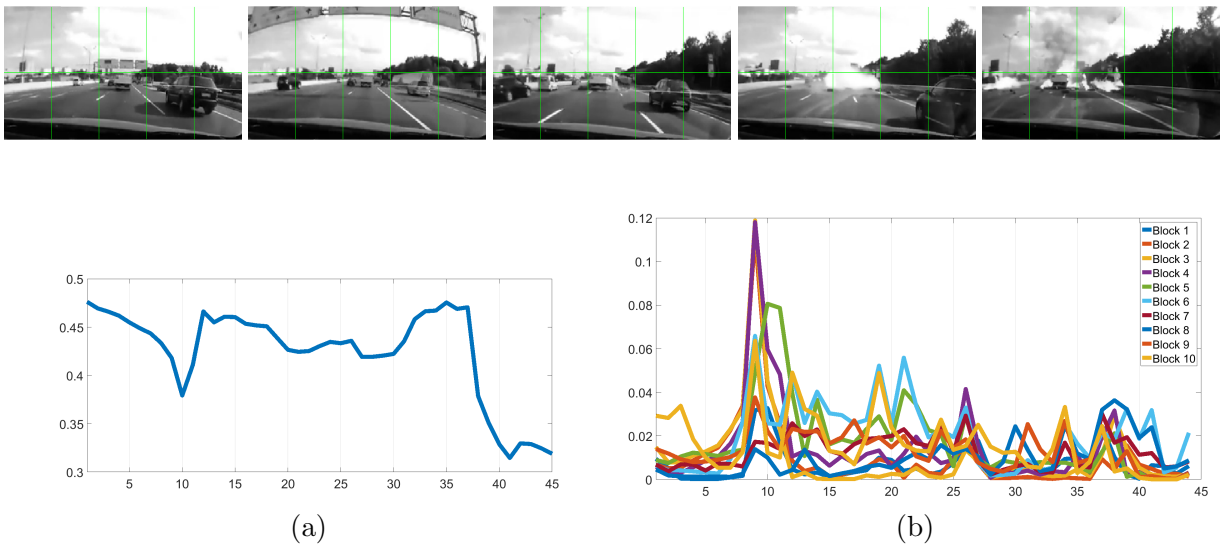


Figure 6.10: The temporal change of intensity profile for each block in 2^{nd} video. (a) shows the temporal change considering the entire frame, (b) shows the temporal change occurring for each block. The x-axis denotes time for both figures. The event occurs at 9th frame in this case.

or entirely. Hence, designing the event as sparse is not appropriate. In this proposed method, we design the temporal set of block features as a summation of two matrices (i) a sparse linear combination of a dictionary which models the foreground or events and (ii) a low rank representation that approximates the background is given by the following equation,

$$\mathbf{Y}_i \approx \mathbf{D}\mathbf{X}_i + \mathbf{B}_i \tag{6.4}$$

Here, each column of $\mathbf{Y}_i \in \mathbb{R}^{p \times N}$, given as $\mathbf{y}_{i,t}$, denotes features from the i^{th} block of t^{th} video frame. The dimension of a feature vector or a block is given by p and N denote the total number of frames in the video. The features for all the blocks in N frames are represented as a linear combination of a dictionary $\mathbf{D} \in \mathbb{R}^{p \times K}$. The coefficients of linear combination are denoted by $\mathbf{X}_i \in \mathbb{R}^{K \times N}$. The low rank matrix for a block is given by $\mathbf{B}_i \in \mathbb{R}^{p \times N}$. In order to identify whether a block contributes to the foreground and background, we add a constraint based on the spatio-temporal saliency of a block. This is based on the hypothesis that a salient block signifies an event, and thereby should be assigned greater importance.



(a) (b)

Figure 6.11: The temporal change of intensity profile for each block in 2^{nd} video. (a) shows the temporal change considering the entire frame, (b) shows the temporal change occurring for each block. The x-axis denotes time in both figures. Here, the event occurs from 26th frame onwards

In the next subsections, we first describe the method for computing the spatio-temporal saliency and then the mathematical derivation of the proposed approach. Finally, we provide the experimental validation using our method and comparison with state of the art methods.

6.3.1 Spatio-temporal saliency

Spatio-temporal saliency, i.e., the 3D(2D+t) equivalent of this problem still needs to be explored. In this work, we employ a contrast based method for spatio-temporal saliency detection. Let c_i be the mean color of block i and p_i denote the centroid of the block. $\mathcal{E}(i, j, t_2)$ denotes an edge between block i and block in j of frame t_2. We define the edge weights as follows:

$$\mathcal{E}(i, j, t_2) = \begin{cases} 1 & \text{if } (j, t_2) \in \mathbb{N}_i \\ 0 & \text{otherwise} \end{cases} \tag{6.5}$$

The neighborhood $\mathbb{N}_i$ of a block i is shown in Fig. 6.12. The neighborhood consists of the blocks in the same location along time and their first order neighbors. The adjacent spatial

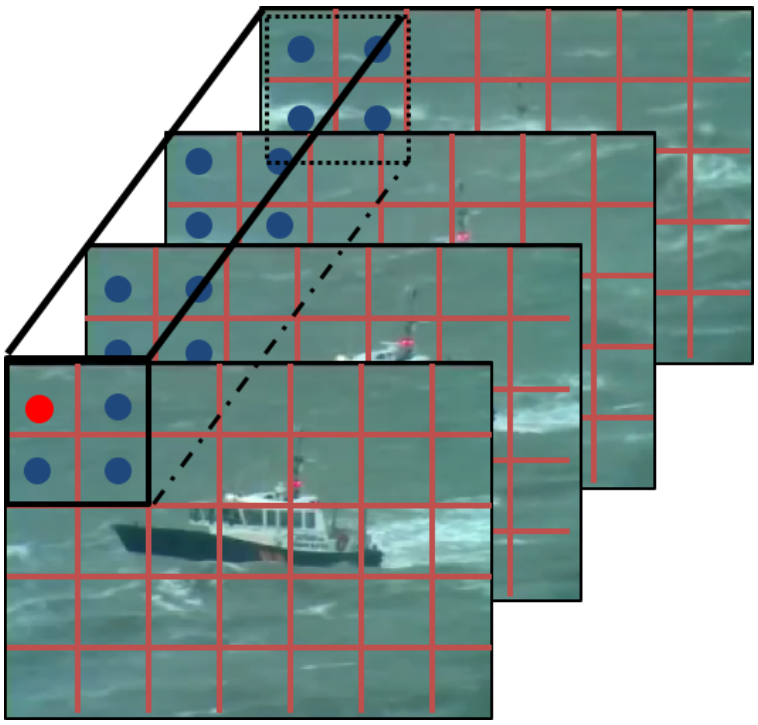


Figure 6.12: The figure shows the neighborhood of a block in computing spatio-temporal saliency. The cuboid formed using blocks with *blue* centroid shown in the upper left corner constitute the space-time neighborhood of the block with *red* centroid.

blocks are defined as first order neighbors. The saliency values for the blocks are computed using,

$$s_i = \sum_{t_2=1}^{N} \sum_{j=1}^{M} \|c_i - c_{j,t_2}\|_2^2 exp(-\|p_i - p_{j,t_2}\|_2^2) \mathcal{E}(i,j,t_2) \tag{6.6}$$

The weights are normalized per block and defined as $w_{i,t_1} = \frac{s_{i,t_1}}{\sum_{t_1=1}^{N} s_{i,t_1}}$, where s_{i,t_1} is the saliency for block i at frame t_1 obtained using (6.6). We denote $(.)_{i,t_1}$ as values corresponding to i^{th} block in t^{th} frame.

6.3.2 Sparse and low rank representation

As shown in (6.4), we model the temporal extent of each local region or block as a combination of a low rank matrix and a sparse linear combination of a dictionary. We need to optimize for the low rank approximation, as well as the dictionary and the sparse code. The optimization problem is defined as follows,

$$\min_{\mathbf{D},\mathbf{X},\mathbf{B}} \sum_{i=1}^{N} f_i(\mathbf{D}, \mathbf{X}_i, \mathbf{B}_i)$$

We define,

$$f_i(\mathbf{D}, \mathbf{X}_i, \mathbf{B}_i) = \|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i - \mathbf{B}_i\|_F^2 + \|\mathbf{B}_i\|_* + \|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i\|_F^2 \mathbf{W}_i + \sum_{t_1=1}^{N} \|\mathbf{x}_{i,t_1}\|_1 \qquad (6.7)$$

Here $\mathbf{W}_i$ denotes a diagonal matrix, for which the diagonal entries contain the normalized saliency values $\mathbf{W}(i,i) = w_i$

The dictionary $\mathbf{D}$ is learned from all the features in the entire video, while the sparse codes and the low rank matrix are computed for each block separately. To solve the problem defined in (6.7), we perform the optimization as an alternating minimization process i.e., we sequentially update $\mathbf{X}$, $\mathbf{D}$ and $\mathbf{B}$ respectively. $\mathbf{X}$, $\mathbf{D}$ are initialized by solving the following

$$\min_{\mathbf{D},\mathbf{x}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \qquad \text{s.t.} \qquad \|\mathbf{x}_{i,t_1}\|_0 \leq \tau \,\forall\, i = 1 \ldots M \text{ and } t_1 = 1 \ldots N$$

Here $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_M]$ and $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_M]$. ℓ_0 is the sparsity inducing norm and τ is the upper bound on the sparsity level. The optimization in (6.8) is solved using the K-SVD algorithm [6]. The K-SVD algorithm solves the optimization as a two step process. It first optimizes the function in (6.8) with respect to $\mathbf{X}$ to obtain the sparse codes using the orthogonal matching pursuit algorithm [45]. In the next step, each column of the dictionary is updated using the precomputed sparse codes. The background feature for each block is initialized as $\mathbf{B}_i = \mathbf{Y}_i - \mathbf{D}\mathbf{X}_i$. After initialization, we update $\mathbf{X}$, $\mathbf{D}$ and $\mathbf{B}$ alternatively until a convergence criterion is reached.

Update of sparse code X

The solution of the sparse code is given by fixing the dictionary $\mathbf{D}$ and $\mathbf{B}$ to their previously updated values. To solve for the sparse codes, we have to minimize the objective function

defined in (6.7) with respect to $\mathbf{x}$, i.e., $\min_x g_x(\mathbf{x})$, where

$$\min_{x} \sum_{t_1=1}^{N} \sum_{i=1}^{M} \|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1} - \mathbf{b}_{i,t_1}\|_F^2 + w_{i,t_1}\|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1}\|_F^2 + \|\mathbf{x}_{i,t_1}\|_1 \tag{6.8}$$

Generally, the ℓ_1 norm optimization can be solved using matching pursuit algorithms. Here we propose a to solve the above problem using the alternating direction method of multipliers (ADMM) [167] In [168] the authors used ADMM to solve for the sparse codes with graph regularization. In order to solve for the sparse variable, the sparsity constraint need to be separated from the main variable. We introduce a variable $\mathbf{Z}$, which has same dimension as that of $\mathbf{X}$ and rewrite the function of (6.8) as follows,

$$\min_{x} \sum_{t_1=1}^{N} \sum_{i=1}^{M} \|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1} - \mathbf{b}_{i,t_1}\|_F^2 + w_{i,t_1}\|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1}\|_F^2 + \|\mathbf{z}_{i,t_1}\|_1 \quad \text{s.t.} \quad \mathbf{X} = \mathbf{Z} \tag{6.9}$$

In the optimization, we update each column of $\mathbf{X}$ separately. The augmented Lagrangian form of the above equation can be written as follows

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{X}, \mathbf{Z}, \mathbf{U}) &= \sum_{t_1=1}^{N} \sum_{i=1}^{M} \|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1} - \mathbf{b}_{i,t_1}\|_F^2 + \\ &\quad w_{i,t_1}\|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1}\|_F^2 + \rho\|\mathbf{x}_{i,t_1} - \mathbf{z}_{i,t_1} + \mathbf{u}_{i,t_1}\|_2^2 + \|\mathbf{z}_{i,t_1}\|_1 \end{aligned}$$

$\mathbf{u}_{i,t_1}$ is the scaled dual variable. Each of the variables $\mathbf{X}, \mathbf{Z}$ and $\mathbf{U}$ is updated alternatively given by the update rule.

$$\begin{aligned} \mathbf{x}_{i,t_1}^{(k+1)} &= \min_{\mathbf{x}} \sum_{t_1=1}^{N} \sum_{i=1}^{M} \|\mathbf{y}_{i,t_1} - \mathbf{D}^{(k)}\mathbf{x}_{i,t_1} - \mathbf{b}_{i,t_1}^{(k)}\|_F^2 \\ &+ w_{i,t_1}\|\mathbf{y}_{i,t_1} - \mathbf{D}^{(k)}\mathbf{x}_{i,t_1}\|_F^2 + \rho\|\mathbf{x}_{i,t_1} - \mathbf{z}_{i,t_1}^{(k)} + \mathbf{u}_{i,t_1}^{(k)}\|_2^2 \\ \mathbf{z}_{i,t_1}^{(k+1)} &= \min_{\mathbf{z}} \sum_{t_1=1}^{N} \sum_{i=1}^{M} \rho\|\mathbf{x}_{i,t_1}^{(k+1)} - \mathbf{z}_{i,t_1} + \mathbf{u}_{i,t_1}^{(k)}\| + \|\mathbf{z}_{i,t_1}\|_1 \\ \mathbf{u}_{i,t_1}^{(k+1)} &= \mathbf{u}_{i,t_1}^{(k)} + \mathbf{x}_{i,t_1}^{(k+1)} - \mathbf{z}_{i,t_1}^{(k+1)} \end{aligned} \tag{6.10}$$

Here k is the iteration number. Here $\mathbf{x}_{i,t_1}^{(k+1)}$ can be obtained using a closed form solution, while $\mathbf{z}_{i,t_1}^{(k+1)}$ can be solved using soft-thresholding method and a least square update on the non-zero components of $\mathbf{z}_{i,t_1}^{(k+1)}$. For details of the solution see Appendix C.2.1

Update of dictionary D

The objective of learning a dictionary is to obtain a set of over-complete basis functions, the linear combination of which will provide an approximation of the original signal. The dictionary for the $(k+1)^{\text{th}}$ iteration, $\mathbf{D}^{(k+1)}$, is computed by solving the following minimization problem

$$\min_{D} \sum_{t_1=1}^{N} \sum_{i=1}^{M} \|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1}^{(k+1)} - \mathbf{b}_{i,t_1}^{(k)}\|_F^2 + w_{i,t_1}\|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1}^{(k+1)}\|_F^2 \tag{6.11}$$

Each column of the dictionary is updated separately. A closed form solution for each dictionary atom can be obtained by taking the derivative of the above function and setting it to zero. Details of the optimization is given in Appendix C.2.2.

Update of background B

The background matrix is constructed as a low rank approximation of the difference between the actual feature and the linear combination of a dictionary of a block. As discussed earlier, since we seek the background for the temporal sequence of a block to be low rank, we optimize $\mathbf{B}$ for each block separately, i.e., $\forall i = 1, 2 \ldots M$, the optimization is described by,

$$\min_{\mathbf{B}_i} \|\mathbf{Y}_i - \mathbf{D}\mathbf{X}_i - \mathbf{B}_i\|_F^2 + \|\mathbf{B}_i\|_* \tag{6.12}$$

Here $\mathbf{Y}_i \in \mathbb{R}^{p \times N}$ is a sub-matrix of $\mathbf{Y}$ containing the features on block i along time. $\mathbf{X}_i \in \mathbb{R}^{K \times N}$ are the corresponding sparse codes. $\|.\|_*$ denotes the nuclear norm for low lank matrix completion. We use the soft-impute algorithm [169] to optimize the equation given in (6.12).

6.4 Event detection from video using SpLoRed

The event detection using **SpLoRed**, is performed by analyzing the error for low rank reconstruction of the background. In the method, each block can be thought of as a combination of foreground and background. In certain blocks where no event occurs, the blocks are reconstructed entirely as a background while the blocks where actual events occur are reconstructed as foreground. Based on this idea of the SpLoRed algorithm, we analyze the background reconstruction error of each block to identify the occurrence of an event.

Temporal detection of events

Event detection for the proposed algorithm can be analyzed in two ways. One can either check the minimum reconstruction of a block with its foreground which is given by the sparse linear combination of the learned dictionary, or based on where the maximum change in the background occurs. The salient idea of this work is to reconstruct the blocks which are more probable to contain event as a linear combination of a learned dictionary. The difference of the block and the low rank reconstruction of the background gives a notion of the event occurring in each block

A measure of to detect the temporal change in a video due to an event in a frame can be obtained by analyzing the difference between the block and the low rank reconstruction of the background, which is given by $\|\mathbf{y}_{i,t_1} - \mathbf{b}_{i,t_1}\|_2^2$ for block i. To consolidate this idea for a particular frame, the maximum error over all the blocks for a that frame is computed and as follows,

$$\dot{\epsilon}(t_1) = \max_i \|\mathbf{y}_{i,t_1} - \mathbf{b}_{i,t_1}\|_2^2 \tag{6.13}$$

The temporal occurrence of an event is identified by thresholding $\dot{\epsilon}(t_1)$ in the similar manner as done in SSPARED [21]. The threshold is chosen by taking the maximum value of $\dot{\epsilon}(t_1)$ for the first 5 frames i.e., $t = 1...5$.

Table 6.2: Confusion matrix (%) for event detection in video

		Event	No Event
Event	SSPARED [21]	**100**	0
	SpLoReD	**100**	0
	ADM [1]	60	40
	DRMF [2]	60	40
No Event	SSPARED [21]	5.8	94.2
	SpLoReD	2.7	**97.3**
	ADM [1]	15.8	84.2
	DRMF [2]	13.9	86.1

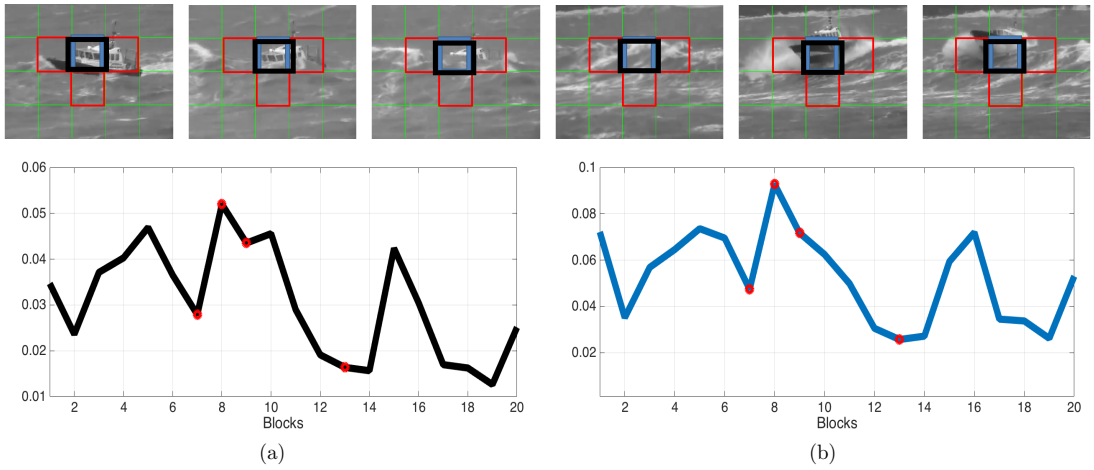


Figure 6.13: The top row shows sample frames from videos. In (a) we plot $\tilde{\epsilon_b}$ and in (b) we plot $\tilde{\epsilon_{bf}}$. The ground truth is given by red. In the top row the ground truth is marked by red. The blocks detected by (a) and (b) are marked by black and blue on the frames respectively

The experiments were performed on the four videos. For video 1, SpLoRed detect the event but has a false positive of 8.6%. For the 2^{nd} and 4^{th} video, our method detects the event without any false positive. For the 3^{rd} video the method yields a false positive of 3.5%. While detecting an event correctly is important, lower level of false positive is equally crucial.

The overall performance over all the four videos is consolidated in table 6.2. The event detection results for the comparison algorithms are also shown in table 6.2. As is noticed from the confusion matrix in table 6.2, SpLoRed and sspared perform equally well in detecting an event, but SpLoRed reduces false positive in identifying events.

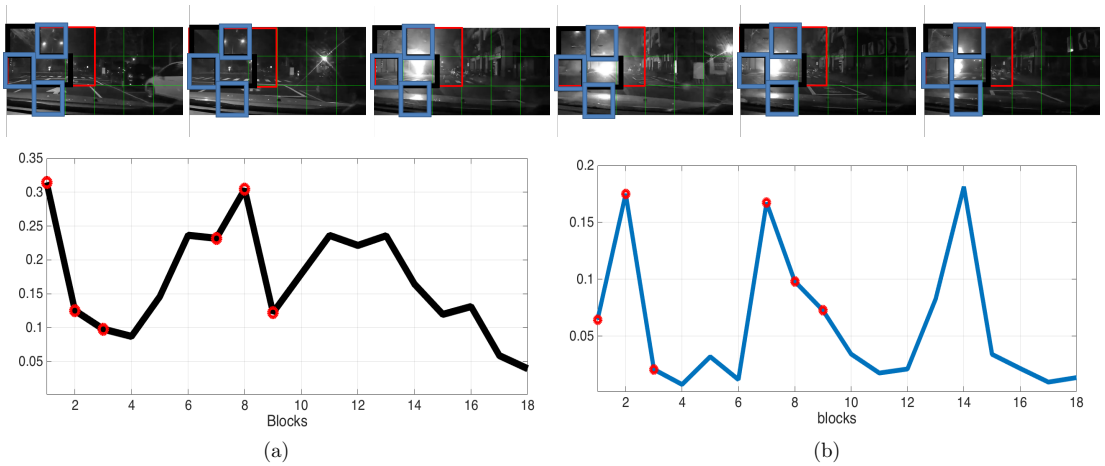


Figure 6.14: The top row shows sample frames from videos. In (a) we plot $\tilde{\epsilon_b}$ and in (b) we plot $\tilde{\epsilon_{bf}}$. The ground truth is given by red. In the top row the ground truth is marked by red. The blocks detected by (a) and (b) are marked by black and blue on the frames respectively

Spatial localization of events

An advantage of the algorithm proposed this section, is that by analyzing the errors for each block, the events can be localized spatially as well, in contrast to SSPARED. In Fig. 6.13 and 6.14, the blocks which spatially exhibit the event are marked in red.

To spatially locate the events, we first plot the mean error of the background along time and a combination of the background error and the foreground error. The mean background error is given by $\tilde{\epsilon_b}(i) = \frac{1}{N}\sum_{t_1=1}^{N} \|\mathbf{y}_{i,t_1} - \mathbf{b}_{i,t_1}\|_2^2$. The mean foreground error is given by $\tilde{\epsilon_f}(i) = \frac{1}{N}\sum_{t_1=1}^{N} \|\mathbf{y}_{i,t_1} - \mathbf{D}\mathbf{x}_{i,t_1}\|_2^2$.

In Fig. 6.13(a), we plot $\tilde{\epsilon_b}$ (black) and in Fig. 6.13(b), we plot $\tilde{\epsilon_{bf}} = \tilde{\epsilon_b} + \tilde{\epsilon_f}$ (blue). The ground truth is marked in red. The maximum values of the errors are used to spatially localize the events. It can be noticed from the graphs that in both scenarios the with maximum value of $\tilde{\epsilon_b}$ the central block can be identified correctly. On the other hand, with $\tilde{\epsilon_{bf}}$, the peaks are more prominent, but for the second video, it generates a false positive.

Discussion

In this proposed method, we have developed an algorithm to detect hazardous events from temporal sequence by a volumetric analysis of the blocks extracted from the video. The proposed solution exploits the spatio-temporal saliency values to learn a sparse representation which provides an estimate of foreground. A low rank approximation, on the other hand, is performed to estimate the background. The approach can be employed to identify the temporal occurrence of the event as well as spatially localize the event. As observed from the experimental results, SpLoRed performs significantly better than the competing methods in detection the time of occurrence of the event as well as identifying the spatial region where the event occurs.

Chapter 7

Conclusion and future work

In image processing and computer vision, extracting informative features has been an important aspect in various applications. Extraction of features in traditional image processing usually denote obtaining color, texture, frequency information of the images. With recent advancement of sparse representation and dictionary learning techniques it was shown that the low-level image features like color, texture, *etc.* can be made more distinctive between images. In sparse representation, the image features are represented as a linear combination of an over-complete basis function. This generates a higher dimensional feature, but the sparsity constraint uses only a few elements of the over-complete basis achieving significant compression. This two fold process of representation in higher dimension and simultaneous compression makes the inherent pattern in the data more prominent.

The initial experiments of sparse representation based dictionary learning, where the sparse codes and the over-complete basis for representation are learned simultaneously, was shown to achieve significant improvement over the existing image de-noising algorithms. With advancement in research it was shown that these methods yield noticeable improvement in classification as well as tracking applications. However, a critical aspect of dictionary learning methods in image analysis applications, which still needs to be explored, is the choice of features and how to integrate them in the learning paradigm. For a single image, whether

color, texture, structure or frequency information is a more reliable representation is still a big challenge. Another problem which demands more research is identifying the local image regions which are more informative. In this thesis, we addressed the above mentioned open problems in dictionary learning methods in the application of image classification and event analysis in video.

7.1 Discussion and summary of the proposed works

In this concluding chapter, we summarize each method and discuss the advantages and limitations of each of the proposed method. We also discuss the future directions of these applications. In the preceding chapters, we proposed methods for integrating feature selection with sparse representation techniques to identify more informative features. The proposed methods were designed for image segmentation, classification and event detection from video. In the first application, we demonstrated that for image segmentation, the non-homogeneous intensity profile can be better modeled using available data increasing segmentation accuracy. In the second contribution, we showed a method of nominating more relevant features for a query image to improve image classification. In the third application, we demonstrated that extracting local relevant features from images can be more effective in identifying the more discriminative features from a single image to aid classification. Finally, in our fourth application we showed that the feature selection method can be extended to a 3D framework i.e., analyzing videos to detect unusual events.

A. In Chapter 3, we proposed a novel segmentation algorithm to deal with intensity inhomogeneity present in various imaging techniques. In this work, a novel segmentation method is proposed which combines the idea of dictionary learning and region based segmentation algorithm in presence of significant clutter and heterogeneous intensity. The method was applied to a blood vessel segmentation from ultrasound image for phlebotomy applications. The method can be summarized as follows

- The presence of clutter and noise added with contrast variation yields improper edge information and hence we employ a region based technique for image segmentation.
- The salient idea of the method is that if similar training images are available, the intensity profile can be learned from these training samples.
- In our approach region intensities are modeled as linear combination of columns of learned dictionary. The dictionary *atoms* act as the 'detail functions' in addition to the mean constant intensity of an image.
- In the region based segmentation method, instead of using two constant intensities to separate the regions, our method uses a linear combination of the learned dictionaries to model the two region intensities.

Our method outperforms other region based methods such as Chan-Vese [30], L2S [54], *etc.*. The images obtained are a sequence of depth images of blood vessels, hence depending on the depth, the image intensity varies significantly. However, if training images of this depth sequence are available, the intensity profile can be modeled using dictionary learning methods. From the experimental validation, it is observed that DL2S outperforms the state of the art in terms of handling heterogeneous image intensity, contour initialization and demonstrates accurate segmentation in cluttered images without the use of explicit shape priors. Moreover, we achieve significant improvement in segmentation accuracy when using learned basis function in comparison to using pre-defined basis function. In our application, the data available is pre-registered. In scenarios where pre-registered data is unavailable, a preprocessing step is needed prior to learning the dictionary.

B. In Chapter 4 we proposed the **meta-algorithm** for feature nomination. The method can be summarized as follows:

- The **meta-algorithm** employs a discriminative dictionary learning based classification scheme and an information theoretic feature nomination algorithm to automatically

decide the most discriminative feature type, from a pool of features, for the query/test image.

- For a test image, initially, classification decision is made using each of these feature types. Next, feature nomination is performed such that, from a pool of features descriptors, the query-specific discriminative feature can be identified.
- The class label for the query for the different feature descriptors and the corresponding sparse representation is obtained using discriminative dictionary learning. The sparse representations of the query and that of the class identified by dictionary learning are analyzed to finalize which feature type is more informative about the query. The final classification is done based on the nominated feature type.

We showed experimental results on the Caltech 101 [170] dataset and achieved significant improvement over bagging method of classifier selection. The discriminative dictionary learning employed in the meta algorithm uses a linear classifier model which may not be appropriate for different datasets. Generally, in complicated datasets as in images, the different classes are often not linearly separable. In such scenarios, the meta-algorithm fails to perform adequately. This problem can be addressed by non-linearly transforming the data (see learning the dictionary from the nonlinearly transformed data as discussed in A). We used only four different types of feature descriptors, whereas in literature a vast pool of feature types exists. Additionally, the **Meta-algorithm** extracts the features from the images by prioritizing all the regions in the image uniformly i.e., does not account for the object of interest in the image. In applications where pre-annotated data is limited, obtaining an over-complete dictionary is not feasible.

C. To deal with scenarios where training data is limited, we propose the method of saliency guided dictionary learning **SDL**. With a small sized training dataset, learning a classifier often leads to over fitting thus increasing the mis-classification rate. In **SDL** and **SDLs**

in Chapter 5, we propose a method for selecting local discriminative features to obtain a more robust representation of the images. The method **SDL** is summarized as below.

- The method integrates the salient region detection technique with sparse coding based dictionary learning. The idea is to learn compact representation of a single image by leveraging the saliency values of local regions such that the more salient regions contribute more towards learning the dictionary.
- The learned sparse codes were employed in computing a similarity measure between pair of images which was further used in the application of classification
- We designed two different similarity measures exploiting the sparse codes. The first involved analyzing the compressibility of the sparse codes, when a pair of images are represented with respect to each other's dictionary.
- The second method designs a sparse code histogram using the cross-dictionary representation and uses K-L divergence method to compare between images.

SDL uses a static, local contrast based saliency detection technique to identify relevant image regions. These local saliency detection techniques do not employ any smoothness constraint and hence the intra object regions and not uniform. The saliency is higher towards the boundary regions of the object. However, to extract information about an object, we require intra-object regions as well. To address this issue, we develop **SDLs**, which employs a smoothness constraint to obtain the saliency map along with the dictionary update. The method can be summarized as below.

- **SDLs** employs a graph based smoothness criterion to obtain a smoother saliency map for an image.
- The saliency map is updated as we update the dictionary, to incorporate the reconstruction error i.e., a region with high reconstruction error should not have a high saliency value.

- The sparse codes obtained using **SDLs** were also used in conjunction with the compression based and sparse code histogram based similarity measure for image classification

As mentioned earlier, we developed a similarity measure for comparing image pairs by leveraging the more discriminative image regions. The goal was to exploit the similarity measure in image classification where training data is limited. Histo-pathological tissue image classification is a classical example where obtaining pre-annotated dataset is challenging. **SDL** and **SDLs** were employed in tissue image classification and also for military vehicle recognition. Both methods achieved significant improvement over the state of the art.

One of the major issues with the method is that it requires comparing image pairs which is computationally expensive. However, since the matching part is independent of other images, it can be parallelized to great extent. The histogram used for similarity measure is obtained by consolidating the local sparse codes. In certain cases of histological tissue classification, local analysis is required. In such scenarios analyzing the local sparse codes are more desirable.

The applications of saliency based dictionary learning is not limited to image classification and can be extended to video analysis problems for identifying events.

D. In Chapter 6, we extend the saliency based dictionary learning in analyzing rare unusual, specifically hazardous events from video. We devise two methods for detecting unusual events from videos **SSPARED** and **SpLoRed**. The method developed in **SSPARED**, analyses consecutive frames to detect events from videos. The method is summarized as follows,

- **SSPARED** employs the saliency based dictionary learning technique similar to **SDL**
- The features of consecutive frames are analyzed using the cross-dictionary representation. The K-L divergence of sparse code histograms of consecutive frames provide a measure of change of the histograms.

- The change in the K-L divergence is tracked to identify the temporal occurrence of the event.

This method involves frame by frame analysis of the video to identify the occurrence of the event in the video. Comparing every frame in a video is again computationally expensive. Moreover, the saliency detection is performed per frame basis and does not involve any temporal information. Since regions can be more relevant in space and time, the temporal transition of these local regions need to be incorporated.

To deal with this, we designed **SpLoRed**, which involves a volumetric analysis of the video to analyze and detect an event. The method is summarized as follows,

- In **SpLoRed** unlike **SSPARED**, we extract spatio temporal saliency by analyzing the video as a block.
- In addition to integrating saliency guided dictionary learning we introduce a low rank representation to identify the regions which has no events. The saliency provides an information about regions with probable events.
- In this problem, each frame is represented as a combination of sparse representation based dictionary learning and a low rank representation. The low rank matrix represents the background of the blocks while the sparse linear combination of the dictionary *atoms* represents the foreground.
- Analyzing both low rank background representation and the foreground provide an information about the spatial and temporal location of the event.

Both **SpLoRed** and **SSPARED** aid in analyzing videos to detect rare and hazardous events and achieve significantly better results in comparison to other background subtraction based method for event or change detection. However, **SpLoRed** unlike **SSPARED** performs a volumetric analysis of the video. This is advantageous since it is a more global method in comparison to frame by frame methods. **SSPARED** is susceptible to local

changes occurring due to occlusion, camera jitter, *etc.* These problems can be overcome using **SpLoRed**.

In this section, we discussed the summary of each of the proposed methods and their limitations. In the next section, we will discuss the future prospects for the methods and applications.

7.2 Concluding remarks and future works

In this work, we developed sparse representation based dictionary learning integrated with relevant feature detection method. We further employed the sparse codes in devising a similarity measure to compare a pair of images. We demonstrated the developed algorithms in applications of image segmentation, classification and event detection.

Our approach for image segmentation, as evident from the quantitative and qualitative experimental results, is capable of handling inhomogeneous intensity better than other region based segmentation methods. However, in our experiments, we performed the training for obtaining intensity profile on a *leave one out* basis. An ideal extension of this approach would be to train the intensity profile on phantom images and test on real time images.

We validated the efficacy of our classification algorithm in histo-pathological tissue image classification as well as other natural image classification applications. We further showed that the developed method for image similarity is not limited to the application of image classification and can be extended to video analysis. The similarity measure developed here can also be exploited in unsupervised methods as in clustering. Since both **SDL** and **SDLs** perform a similarity between a pair of images and do not incorporate any class label information in learning the sparse codes unlike the **meta-algorithm**. Hence this the method developed in this work can be exploited for unsupervised classification when class labels not present.

Additionally, the sparse representations obtained from **SDL** and **SDLs** can be exploited as image features in conjunction with other classification techniques to handle larger datasets. In our experiments, we used the local Gabor filter response as feature descriptors for the images. Gabor features are appropriate for extracting texture information from the images but to extend the work from tissue to any natural image classification, using just Gabor features is not sufficient. An ideal extension of this work would be integrating the meta-algorithm for feature nomination with the **SDLs**. This will allow us to exploit the advantages of both local feature selection as well as identifying the more relevant feature descriptors to boost the classification framework.

Furthermore, for the rare and unusual event detection application, the method identifies the frames or time of occurrence of the events. At this point the algorithm exploits the saliency and sparse representation techniques to detect events. However, the method is not capable of identifying between hazardous and non-hazardous events for a more practical surveillance. To provide a complete solution for such a problem, an integrated method for detection and recognition which can differentiate between hazardous and non-hazardous events needs to be incorporated.

7.3 Publication list resulting from this work

1. **Rituparna Sarkar** and Scott T. Acton "SpLoRed:Spatio-temporal saliency guided sparse and low rank representation", (in preparation)

2. **Rituparna Sarkar** and Scott T. Acton "SDL: Saliency guided dictionary learning for image similarity", (under review, IEEE Transactions in Image Processing)

3. Tamal Batabyal, **Rituparna Sarkar**, Scott T. Acton "GraDED: A graph-based parametric dictionary learning algorithm for event detection" (submitted, IEEE International Conference on Image Processing (ICIP) 2017)

4. Tiffany T. Ly, **Rituparna Sarkar** and Scott T. Acton "IP on AP:Image processing on Automata processor", (submitted, IEEE International Conference on Image Processing (ICIP) 2017)

5. Jie Wang, **Rituparna Sarkar**, Arslan Aziz, Andrea Vaccari, Andreas Gahlmann, Scott Acton "BACT-3D: A level set segmentation approach for dense multi-layered 3D bacterial biofilms", (submitted, IEEE International Conference on Image Processing (ICIP) 2017)

6. Tiffany T. Ly, **Rituparna Sarkar**, Scott T. Acton and Kevin Skadron "Feature Extraction and Image Recognition from Superpixels on an Automata Architecture", IEEE Asilomar Conference on Signals, Systems and Computers 2016

7. **Rituparna Sarkar** and Scott T. Acton "SLIDE: Saliency guided image dictionary and image similarity evaluation",IEEE International Conference on Image Processing IEEE International Conference on Image Processing (ICIP), 2016

8. **Rituparna Sarkar** Andrea Vaccari and Scott T. Acton "SSPARED: Saliency and sparse code analysis for rare event detection in video", IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), 2016

9. **Rituparna Sarkar** Suvadip Mukherjee and Scott T. Acton "Dictionary Learning Level Set", IEEE Signal Processing Letters, 2015.

10. **Rituparna Sarkar**, Sedat Ozer, Kevin Skadron and Scott T. Acton, "Image classification by multi-kernel dictionary learning", IEEE Asilomar Conference on Signals, Systems and Computers, 2014.

11. **Rituparna Sarkar**, Kevin Skadron and Scott T. Acton, "A meta-algorithm for classification by feature nomination" IEEE International Conference on Image Processing (ICIP), 2014.

12. Suvadip Mukherjee,**Rituparna Sarkar**,Joshua Vandenbrink, Scott T. Acton and Benjamin Blackman, "Tracking Sunflower Circumnutation using Affine Parametric Active Contours", IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI), 2014.

13. **Rituparna Sarkar**, Suvadip Mukherjee and Scott T. Acton, "Shape Descriptor Based On Compressed sensing with Application to Neuron Matching", IEEE Asilomar Conference on Signals, Systems and Computers, 2013.

14. **Rituparna Sarkar**, Namrata Vaswani and Samarjit Das, "Tracking Sparse Signal Sequences from Nonlinear/Non-Gaussian Measurements and Applications in Illumination-Motion Tracking", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013.

Appendices

Appendix A

Multi-kernel dictionary learning

The work proposed in the meta-algorithm for feature nomination in chapter 1 show the efficacy of selecting and combining the salient features from a pool of feature types in image retrieval and classification applications. The sparse representation based classification techniques have been developed on the linear representation of the data. But as stated earlier, these methods fail to capture the non-linearity present in the datasets. To deal with non-linearity in the data, the Kernel trick is applied which non-linearly transforms a data to a higher dimensional space. In [89, 90] it has been shown that sparse representation based kernel learning has proved to be efficient for classification purpose. In [171], kernel representation based dictionary learning has shown to efficiently capture the non-linearity in the data and at the same time give a compact representation of the data in the kernel space.

In [12], we demonstrated that for robust classification, choice of feature is of significant importance. A single feature cannot discriminatingly represent all the images in the dataset. Hence we developed a method in which a classifier is designed in a discriminative dictionary learning framework for each of the different feature types. For a test image, the most appropriate feature type is chosen based on the class conditional entropy with respect to the features. One drawback of this method is that it identifies only one feature type as the most significant feature. However in practice we have seen that the nominated feature may not be

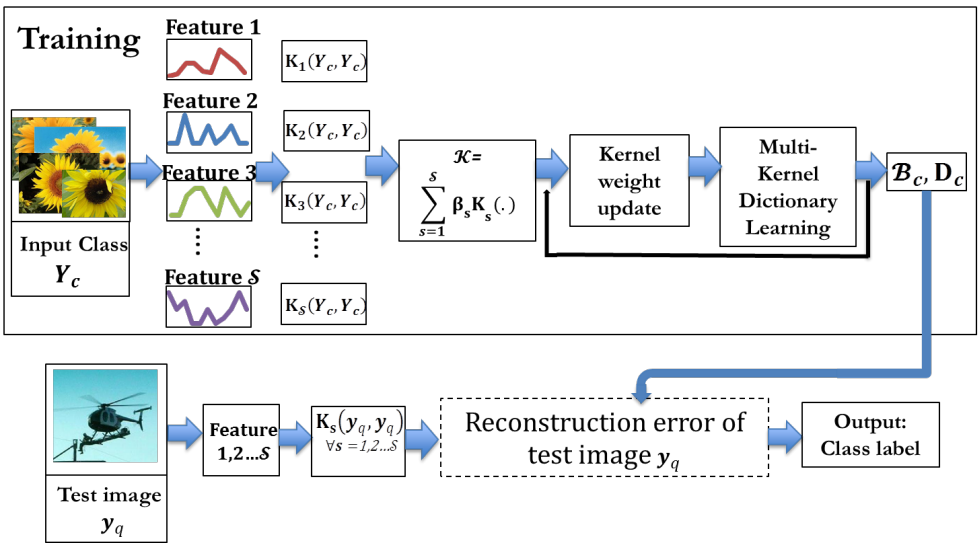


Figure A.1: Overview of the MKDL algorithm

the most discriminative feature for a particular query. This drawback has motivated us to develop a feature combination based dictionary learning framework for robust classification.

Motivated by this fact, we develop an information theoretic kernel combination method embedded in the dictionary learning framework. One advantage of the kernel space representation, other than a higher-dimensional representation, lies in the fact that different features can be combined in the kernel space using multiple kernel functions [172]. The kernel-sparse representation techniques [89, 90] mainly deal with a single kernel in sparse representation or dictionary learning framework. Our method learns a dictionary in the kernel space for each of the classes in the learning phase. We employ a mutual information based approach to obtain the most desirable weights for kernel combination. In the testing phase, our method exploits the learned dictionaries and the kernel weights to assign a class label to the test. The steps involved in the classification system are shown in the block diagram (see Fig. A.1). The training phase involves discriminative feature and the respective kernel matrix computation. The next step in training exploits a combination of these features for a classifier design using a dictionary learning framework. The testing part involves extracting similar types of features as used for training part and using the learned to identify the class for the test image.

A.1 Kernel dictionary learning

Let $\hat{\phi} : \mathbb{R}^N \rightarrow \mathbb{H}$ be the non-linear transformation that transforms the features to a higher dimensional kernel space referred to as reproducing kernel Hilbert space (RKHS). The data $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_c}]$ can now be represented as $\hat{\phi}(Y) = [\hat{\phi}(\mathbf{y}_1), \hat{\phi}(\mathbf{y}_2), \ldots, \hat{\phi}(\mathbf{y}_{N_c})]$ and the kernel similarity function can be defined as $\mathscr{K}(\mathbf{y}_i, \mathbf{y}_i) = \hat{\phi}(\mathbf{y}_i)^T \hat{\phi}(\mathbf{y}_i)$. Similar to sparse representation in feature space, the test data can be represented as a linear combination of the nonlinearly transformed training data [36, 88–90]. Similarly, the dictionary learning framework can be adapted in the non-linear kernel feature space. Let $\mathbb{D}$ be denoted as the dictionary in the kernel-feature space.

$$\min_{\mathbb{D},\mathbf{X}} \|\hat{\phi}(\mathbf{Y}) - \mathbb{D}\mathbf{X}\|_2^2 \quad s.t \quad \|\mathbf{x}_i\|_0 \leq T \quad \forall i = 1 \ldots N_c \tag{A.1}$$

(A.1) solves for the dictionary as well as the sparse codes for representing the non-linearly transformed data, which can also be written as

$$\begin{aligned} & \min_{\mathbf{D},\mathbf{X}} \|\hat{\phi}(\mathbf{Y}) - \hat{\phi}(\mathbf{Y})\mathbf{D}\mathbf{X}\|_2^2 \quad s.t \quad \|\mathbf{x}_i\|_0 \leq T \quad \forall i = 1 \ldots N_c \\ \approx \quad & \min_{\mathbf{D},\mathbf{x}_i} \sum_{i=1}^{N_c} \|\hat{\phi}(\mathbf{y}_i) - \hat{\phi}(\mathbf{Y})\mathbf{D}\mathbf{x}_i\|_2^2 \quad s.t \quad \|\mathbf{x}_i\|_0 \leq T \end{aligned}$$

Where $\mathbb{D} = \hat{\phi}(\mathbf{Y})\mathbf{D}$ and by optimizing for $\mathbf{D}$ and x_i, the desired dictionary can be obtained. The objective function for the reconstruction error is written in terms of kernel functions as,

$$\begin{aligned} \|\hat{\phi}(\mathbf{y}_i) - \hat{\phi}(\mathbf{Y})\mathbf{D}\mathbf{x}_i\|_2^2 &= \hat{\phi}(\mathbf{y}_i)^T\hat{\phi}(\mathbf{y}_i) - 2\mathbf{x}_i^T\mathbf{D}^T\hat{\phi}(\mathbf{Y})^T\hat{\phi}(\mathbf{y}_i) + \mathbf{x}_i^T\mathbf{D}^T\hat{\phi}(\mathbf{Y})^T\hat{\phi}(\mathbf{Y})\mathbf{D}\mathbf{x}_i \\ &= \mathscr{K}(\mathbf{y}_i, \mathbf{y}_i) - 2\mathbf{x}_i^T\mathbf{D}^T\mathscr{K}(\mathbf{Y}, \mathbf{y}_i) + \mathbf{x}_i^T\mathbf{D}^T\mathscr{K}(\mathbf{Y}, \mathbf{Y})\mathbf{D}\mathbf{x}_i \end{aligned}$$

With pre-specified kernel function $\mathscr{K}$, the optimization problem can be solved without any prior knowledge of the non-linearity in the data space.

A.2 Feature combination by multikernel dictionary learning

An advantage of kernel space representation of features is, combination of different features is possible using multiple kernel functions. The goal is to exploit the discriminative property of different features and combine them to achieve a robust classification system. The kernel function in A.1 can be written as a weighted sum of multiple kernel functions. $\mathcal{K} = \sum_{s=1}^{\mathcal{S}} \beta_s \mathscr{K}_s$ $\mathcal{K}$ is the kernel function obtained from a linear combination of $\mathcal{S}$ different kernel functions for $\mathcal{S}$ different features. If f denote function transforming the feature to the kernel feature space, we can re-write the entropy for the transformed feature as $\mathscr{H}(c|f(x))$. The weights for the kernel combination are hence obtained by minimizing the conditional entropy for a class,

$$\min_{B} \mathscr{H}(c|\mathscr{K} = f(\mathbf{Y}, B)) \quad s.t \quad \sum_{s=1}^{\mathcal{S}} \beta_s = 1 \text{and} \quad \beta_s \geq 0 \quad \forall \quad s \tag{A.2}$$

The problem discussed in chapter 4 exploits a dictionary learning based linear classifier each of the different feature types and then chooses the most discriminative feature for classification using the class conditional entropy. Here we address the feature combination problem by multi-kernel dictionary learning.

The training phase involves learning the kernel weights $B_c = [\beta_1 \dots \beta_S]$, the dictionary $\mathbf{D}_c$ and the sparse codes $\mathbf{X}_c$ for each of the classes $c = 1 \dots C$. We define the linear combination of the kernel matrix $\mathcal{K} = \sum_{s=1}^{\mathcal{S}} \beta_s \mathscr{K}_s(.)$such that $\Psi(.)^T \Psi(.) = \mathcal{K}$. Ψ is defined as the non-linear transform from feature space to kernel feature space. The learning phase involves an alternating minimization approach.

i Optimization for B_c: with fixed $\mathbb{D}$ We solve , to obtain the kernel weights. We initialize $\beta_s = \frac{1}{S} \quad \forall s \in [1 \dots S]$, such that $\sum_{s=1}^{\mathcal{S}} \beta_s = 1$.To solve for B_c, we use a random search method [173]. We randomly select weight values from a Gaussian distribution such that, $\beta_s^t \sim \mathcal{N}(\beta_s^{t-1}, 1)$ and normalized by $\sum_{s=1}^{\mathcal{S}} \beta_s$ Then select the β_s values for which

$\mathscr{H}(c|\mathscr{K} = f(\mathbf{Y}, B))$ is minimum. t denotes the iteration step.

ii Updating dictionary $\mathbb{D}$ and sparse codes $\mathbf{X}$: B_c is kept fixed, $\mathbb{D}_c$ is initialized by randomly selecting K columns of $\mathbf{Y}_c$. First keeping $\mathbb{D}_c$ fixed, we update the sparse codes $\mathbf{X}_c$ at time step t solving the following

$$\min_{\mathbf{X}_c} \|\Psi(\mathbf{Y}_c) - \mathbb{D}_c^{t-1}\mathbf{X}_c\|_F^2 \quad \text{s.t} \quad \|\mathbf{x}_i\|_0 \leq T \quad \forall i \in 1, \ldots, N_c \tag{A.3}$$

We use orthogonal matching pursuit [45] to solve eq. (A.3). Once the sparse codes are obtained the next step is to update the dictionary. Keeping the sparse codes fixed, we solve the following equation, with the constraint that the columns of the dictionary will be orthonormal. $\min_{\mathbb{D}_c^t} \|\Psi(\mathbf{Y}_c) - \mathbb{D}_c^{t-1}\mathbf{X}_c^t\|_F^2$

The objective function can re-written as eq. (A.2) and the optimized over $\mathbb{D}_c^t$ [171]. We use the K-SVD algorithms [6] for the dictionary update.

A.3 Image classification by feature MKDL

The classification of the test image is performed based on the minimum reconstruction error with respect to the class dictionaries. Once the feature vectors for the query image, $\mathbf{y}_q$ is available, $\forall c = 1 \ldots C$, the kernel combination for the test image is obtained as $\mathcal{K}_c = f(\mathbf{y}_q, B_C)$, such that $\mathcal{K}_c = \Psi_c^T \Psi_c$. The respective sparse codes $\mathbf{x}_q^c$ corresponding to the class dictionary. The test image is identified to belong to the particular class for which the reconstruction error is minimum.

$$(l(\mathbf{y}_q) = c) = \min_c \|\Psi_c(\mathbf{y}_q) - \mathbb{D}_c\mathbf{x}_q^c\|_2^2 \tag{A.4}$$

We performed experiments on Caltech 101 dataset [170]. The dataset has 101 categories with about 9000 images. About 3000 images were used for training. 30 images were chosen at random per class to perform the training. The rest, about 6000 images were used for testing.

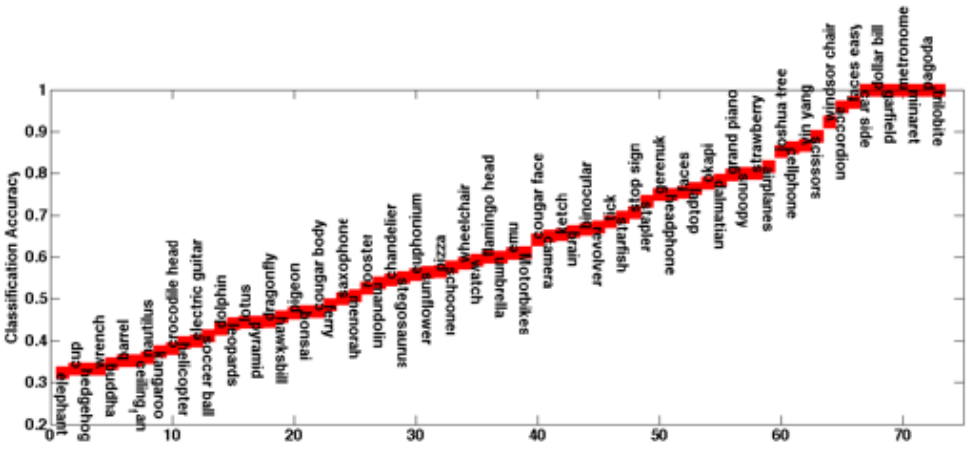


Figure A.2: The figure shows the per class classification accuracy for 70 classes in the dataset.

Figure A.3: Figure shows comparison results with meta algorithm

We performed the experiment with spatial pyramid [139] representation of scale invariant feature transform (SIFT) descriptors [22], with Gaussian $\mathscr{K}(\mathbf{y}_i, \mathbf{y}_i) = exp^{-\gamma\|\mathbf{y}_i - \mathbf{y}_j\|_2^2}$ and polynomial $\mathscr{K}(\mathbf{y}_i, \mathbf{y}_i)) = \mathbf{y}_i^T \mathbf{y}_j$ and histogram of oriented gradients (HOG) [23] descriptors with polynomial kernel. In Fig. A.2, we plot the classification accuracy of 70 classes in ascending order of their classification accuracy. Fig. A.3 shows comparison of classification accuracy of MI-MKDL with meta-algorithm for feature nomination [12] of 24 sample classes.

Discussion

Here, we first introduce a sparse representation based dictionary learning algorithm using kernel space feature representation and then extend this representation by way of multi-kernel learning. The method transforms data into high dimensional feature space to capture the non linearity in the data. The multi-kernel learning allows feature combination and is optimized using mutual information, yielding weights for kernel combination. In contrast to previous work [12], in this paper, we approach this problem as a selection of the salient feature type(s) from a pool of feature types rather than selecting an individual feature from the pool. Our approach utilizes multiple kernels within the dictionary-learning framework where a combination of dictionary atoms represents individual categories. The category specific feature combination parameters or weights for kernel combination are determined by the mutual information techniques.

Appendix B

Analyzing similarity measure for saliency based dictionary with selected images

In this section we show experimental results and effects of dictionary atoms on some hand picked images. This small experiments also demonstrates the effect of object rotation and noise on the similarity measure devised in chapter 5.

B.1 Validation of the proposed method on sample images

To demonstrate the effect of the parameters on similarity measure, we perform experiments on some sample images. These images are obtained from 53 objects database in *http://www.vision.ee.ethz.ch/datasets/*. The experiments were performed on six sample images obtained from the dataset as shown in Fig. B.1. The first two columns (Fig. B.1 (a) and (b)) are the images selected from the dataset with rotated version of the same object. The third and fourth columns (Fig. B.1 (c) and (d)) are the noisy versions of the original

images, obtained by adding Gaussian noise to the original images. The images with blue border are used as the test images. The three rows correspond to the three classes.

We performed experiments using different combinations of dictionary size (K) and number of superpixels (N). The Number of dictionary atoms ranged from $K = 100$ and 200 while the number of superpixels computed were $N = 1200$ and 520. In Fig. B.2 we plot the mean similarity value using **SDL**, **SDLs** and **DL+KLdiv** (marked as DL) for each the image pairs by fixing either K or N. For example when we use class 1 (a) image (Fig. B.1(a) row 1), the first row of Fig. B.2 shows the mean similarity values for each of the othe 9 images in the dataset (Fig. B.1(b,c,d) of rows 1,2 and 3).

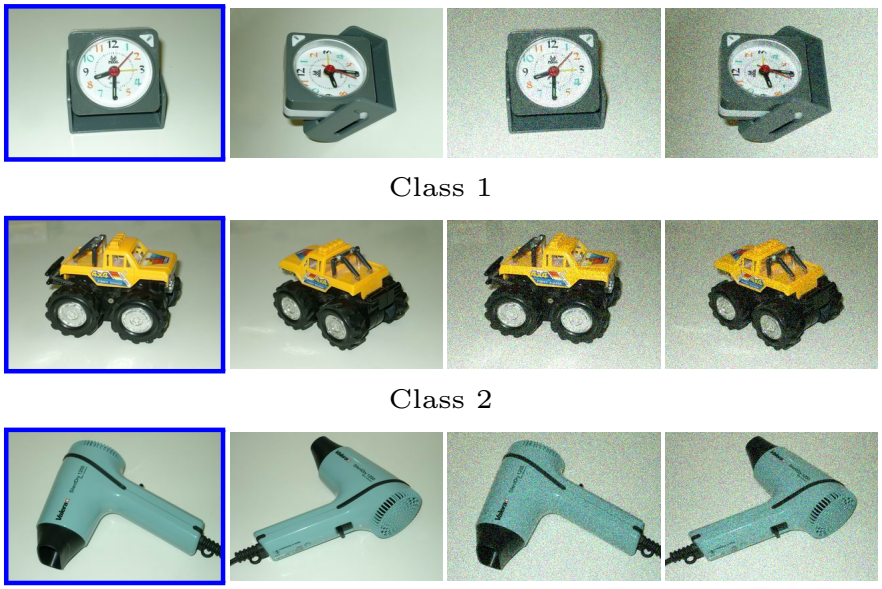


Figure B.1: Sample images for analyzing effect of dictionary and superpixels on the similarity between images. (a) and (b) are rotated versions of the same image obtained from the dataset. (c) and (d) are the original images with Gaussian noise.The images with blue border are used as test images to compare with the others.

Below each of the figure we denote the value of number of dictionary atoms and superpixels. Also below each figure we denote the mean and variation of the similarity values with respect to each of the classes for each of **SDL**, **SDLs** and **DL+KLdiv**. Compared to **SDL** and

Figure B.2: The bar graph plots the mean similarity value and the variance for each combination of test and train image for various parameter value. For e.g., the first bar graph shows the similarity between the test image from class one and the other nine images from the training set when no. of dictionary atoms, $K = 100$ and the variance in the similarity with no. of superpixels, $N = 1200$ and 520. The first three columns in each bar graph correspond to class 1, next three to class 2 and the last three belong to class 3. Below each graph, the mean similarity measure and the variance for each class is given.

DL+KLdiv, **SDLs** has a higher inter-class variance, which is very significant in distinguishing or separating between two classes.

It is also noted from the intra-class variance should be small in addition to higher inter-class variance. From the parameters, it is noted that the above criteria holds for $N = 1200$ with varying K and $K = 200$ with varying N. Hence in our experiments in chapter 5, we use dictionary atoms around $0.2\% - 0.3\%$ os the number of superpixels.

Appendix C

Derivations of the mathematical results

C.1 Derivation of SDLs equation

The update for the saliency weights for the SDLs algorithm is obtained by solving the following equation

$$\begin{aligned}\frac{\partial C}{\partial \mathbf{w}} &= \frac{\partial}{\partial \mathbf{w}}\left\{\sum_{i=1}^{N} w_i\|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 + \sum_{i,j} a_{ij}\|w_i - w_j\|_2^2 + \frac{1}{2}\mu\sum_{i}^{N}\tilde{d}_{ii}\|w_i - \frac{w_0}{\tilde{d}_{ii}}\|_2^2\right\} \\ &= \frac{\partial}{\partial \mathbf{w}}\left\{\mathbf{E}^T\mathbf{w} + \frac{1}{2}\mathbf{w}^T(\tilde{\mathbf{D}} - \mathbf{A})\mathbf{w} + \frac{1}{2}\mu(\mathbf{w} - \tilde{\mathbf{D}}^{-1}\mathbf{w}_0)^T\tilde{\mathbf{D}}(\mathbf{w} - \tilde{\mathbf{D}}^{-1}\mathbf{w}_0)\right\} \qquad \text{(C.1)}\end{aligned}$$

setting $\frac{\partial C}{\partial \mathbf{w}}\big|_{\mathbf{w}^*} = 0$ we get,

$$\begin{aligned}0 &= \mathbf{E}^T + \mathbf{w}^T(\tilde{\mathbf{D}} - \mathbf{A}) + \mu\mathbf{w}^T\tilde{\mathbf{D}} - \mu\mathbf{w}_0^T\tilde{\mathbf{D}}^{-1} \\ \mathbf{w}^T(\tilde{\mathbf{D}} - \mathbf{A}) + \mu\mathbf{w}^T\tilde{\mathbf{D}} &= \mu\mathbf{w}_0^T\tilde{\mathbf{D}}^{-1}\tilde{\mathbf{D}} - \mathbf{E}^T \\ (1+\mu)\mathbf{w}^T(\tilde{\mathbf{D}} - \alpha\mathbf{A}) &= \mu\mathbf{w}_0^T - \mathbf{E}^T \qquad \text{(C.2)}\end{aligned}$$

Since $\tilde{\mathbf{D}}$ is a diagonal matrix, $(\tilde{\mathbf{D}}^{-1})^T = \tilde{\mathbf{D}}^{-1}$. Also $\tilde{\mathbf{D}} - \alpha\mathbf{A}$ is a symmetric matrix matrix, hence $(\tilde{\mathbf{D}} - \alpha\mathbf{A})^T = \tilde{\mathbf{D}} - \alpha\mathbf{A}$ The closed form solution of $w \in \mathbb{R}^{N\times 1}$ can be obtained as

$$\mathbf{w}^* = \beta^{-1}(\tilde{\mathbf{D}} - \alpha\mathbf{A})^{-1}(\mu\mathbf{w}_0 - \mathbf{E}). \tag{C.3}$$

C.2 Derivation of SpLoRed equations

C.2.1 Update of sparse codes

The sparse codes $\mathbf{X}$ is updated column by column and hence can be updated in parallel. The update rule for $\mathbf{X}$ given in eq. (6.10), can be simplified and re-written as,

$$\mathbf{x}_{i,t_1}^{(k+1)} = \min_{\mathbf{x}} \|\mathbf{y}_{i,t_1} - \mathbf{D}^{(k)}\mathbf{x}_{i,t_1} - \mathbf{b}_{i,t_1}^{(k)}\|_F^2 + w_{i,t_1}\|\mathbf{y}_{i,t_1} - \mathbf{D}^{(k)}\mathbf{x}_{i,t_1}\|_F^2 + \rho\|\mathbf{x}_{i,t_1} - \mathbf{z}_{i,t_1}^{(k)} + \mathbf{u}_{i,t_1}^{(k)}\|_2^2$$

A closed form solution for $\mathbf{x}_{i,t_1}^{(k+1)}$ can be obtained by taking the derivative of the above function with respect to $\mathbf{x}$ and setting it to 0. Each column of $\mathbf{X}$ can then be updated using the solution of the following equation.

$$\begin{aligned}(\mathbf{x}_{i,t_1}^{(k+1)})^T[(w_{i,t_1}+1)(\mathbf{D}^{(k)})^T\mathbf{D}^{(k)} - 0.5\rho^2] = \\ [(w_{i,t_1}+1)(\mathbf{y}_{i,t_1}) - \mathbf{b}_{i,t_1}^{(k)}]^T\mathbf{D}^{(k)} - 0.5\rho^2(\mathbf{z}_{i,t_1}^{(k)} - \mathbf{u}_{i,t_1}^{(k)})\end{aligned} \tag{C.4}$$

$(.)^T$ denote the transpose of a variable. The $\mathbf{Z}$ can be updated using the soft thresholding algorithm. Similar to $\mathbf{X}$, we solve for each column of $\mathbf{Z}$ in eq. (6.10) and update using the following equation

$$\mathbf{z}_{i,t_1}^{(k+1)} = \begin{cases} \mathbf{h} - 0.5\lambda & \text{if } \mathbf{h} > 0.5\lambda \\ \mathbf{h} + 0.5\lambda & \text{if } \mathbf{h} < -0.5\lambda \\ 0 & \text{if } -0.5\lambda \geq \mathbf{h} \leq 0.5\lambda \end{cases} \tag{C.5}$$

Here $\mathbf{h} = \mathbf{x}_{i,t_1}^{(k+1)} + \mathbf{u}_{i,t_1}^{(k)}$. The final values of $\mathbf{Z}$ is obtained by a least square update on

the active set of the dictionary. We define the active set as, $\Theta_{i,t_1} = \{\alpha | \mathbf{z}_{i,t_1}^{(k+1)} \neq 0\}$ i.e., the non-zero locations of the vector $\mathbf{z}_{i,t_1}^{(k+1)}$. We update the non-zero components of $\mathbf{z}_{i,t_1}^{(k+1)}$ using $\mathbf{z}_{i,t_1}^{(k+1)} = ((\hat{\mathbf{D}}^{(k)})^T(\hat{\mathbf{D}}^{(k)}))^{-1}(\hat{\mathbf{D}}^{(k)})^T\mathbf{y}_{i,t_1}^{(k+1)}$. $\hat{\mathbf{D}}$ denote the sub-dictionary consisting of the columns in the set Θ_{i,t_1}.

C.2.2 Update of dictionary atoms

Here we update each column of the dictionary at a time. The minimization equation given in eq. (6.11) can be re-written as a function of a single column of $\mathbf{D}$ as follows

$$\sum_{t_1=1}^{N}\sum_{i=1}^{M} \|(\mathbf{y}_{i,t_1} - \mathbf{b}_{i,t_1}^{(k)}) - \mathbf{d}_l(\mathbf{x}_{i,t_1}^{(k+1)})^l - \sum_{m=1\ m\neq l}^{K} \mathbf{d}_m(\mathbf{x}_{i,t_1}^{(k+1)})^m\|_2^2 +$$

$$\|\sqrt{w}_{i,t_1}(\mathbf{y}_{i,t_1} - \mathbf{d}_l(\mathbf{x}_{i,t_1}^{(k+1)})^l - \sum_{m=1\ m\neq l}^{K} \mathbf{d}_m(\mathbf{x}_{i,t_1}^{(k+1)})^m)\|_2^2 \tag{C.6}$$

$$= \|\mathbf{Y} - \mathbf{B} - \mathbf{d}_l(\mathbf{X}^{(k+1)})^l - \sum_{m=1\ m\neq l}^{K} \mathbf{d}_m(\mathbf{X}^{(k+1)})^m\|_F^2 +$$

$$\|(\mathbf{Y} - \mathbf{d}_l(\mathbf{X}^{(k+1)})^l - \sum_{m=1\ m\neq l}^{K} \mathbf{d}_m(\mathbf{X}^{(k+1)})^m)\mathbf{W}^{\frac{1}{2}}\|_F^2 \tag{C.7}$$

$$= \|\mathbb{E}_1 - \mathbf{d}_l(\mathbf{X}^{(k+1)})^l\|_F^2 + \|\mathbb{E}_2 - \mathbf{d}_l(\mathbf{X}^{(k+1)})^l\mathbf{W}^{\frac{1}{2}}\|_F^2 \tag{C.8}$$

Here $\mathbf{d}_{(.)}$ denotes the $(.)^{th}$ column of $\mathbf{D}$, while $\mathbf{X}^{(k+1)})(.)$ denote the $(.)^{th}$ row of $(k+1)^{th}$ iteration of $\mathbf{X}$. The closed for solution for each column of $\mathbf{d}$ can be obtained as follows

$$\mathbf{d}_l = \frac{\mathbb{E}_1((\mathbf{X}^{(k+1)})^l)^T + \mathbb{E}_2\mathbf{W}^T((\mathbf{X}^{(k+1)})^l)^T}{\|((\mathbf{X}^{(k+1)})^l)\|_2^2 + \|((\mathbf{X}^{(k+1)})^l)\mathbf{W}\|_2^2} \tag{C.9}$$

The closed form solution is obtained using the active set Θ_{i,t_1} of $\mathbf{X}$.

Bibliography

[1] A. Adler, M. Elad, Y. Hel-Or, and E. Rivlin, "Sparse coding with anomaly detection," *Journal of Signal Processing Systems*, vol. 79, no. 2, pp. 179–188, 2015.

[2] L. Xiong, X. Chen, and J. Schneider, "Direct robust matrix factorizatoin for anomaly detection," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 844–853.

[3] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *JOSA A*, vol. 4, no. 12, pp. 2379–2394, 1987.

[4] B. A. Olshausen *et al.*, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[5] B. A. Olshausen and D. J. Field, "Sparse coding of sensory inputs," *Current Opinion in Neurobiology*, vol. 14, no. 4, pp. 481 – 487, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959438804001035

[6] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *Image Processing, IEEE Trans. on*, vol. 15, no. 12, pp. 3736–3745, 2006.

[7] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[8] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 689–696.

[9] Q. Zhang and B. Li, "Discriminative k-svd for dictionary learning in face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2691–2698.

[10] R. Sarkar and S. T. Acton, "Slide: Saliency guided image dictionary and image similarity evaluation," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sept 2016, pp. 216–220.

[11] T. Guha and R. K. Ward, "Image similarity using sparse representation and compression distance," *Multimedia, IEEE Trans. on*, vol. 16, no. 4, pp. 980–987, 2014.

[12] R. Sarkar, K. Skadron, and S. Acton, "A meta-algorithm for classification by feature nomination," in *Image Processing (ICIP), 2014 IEEE International Conference on*, Oct 2014, pp. 5187–5191.

[13] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent k-svd," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1697–1704.

[14] U. Srinivas, H. S. Mousavi, V. Monga, A. Hattel, and B. Jayarao, "Simultaneous sparsity model for histopathological image representation and classification," *IEEE Transactions on Medical Imaging*, vol. 33, no. 5, pp. 1163–1179, May 2014.

[15] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.

[16] T. H. Vu, H. S. Mousavi, V. Monga, G. Rao, and U. A. Rao, "Histopathological image classification using discriminative feature-oriented dictionary learning," *IEEE transactions on medical imaging*, vol. 35, no. 3, pp. 738–751, 2016.

[17] R. Sarkar, S. Mukherjee, and S. T. Acton, "Dictionary learning level set," *Signal Processing Letters, IEEE*, vol. 22, no. 11, pp. 2034–2038, 2015.

[18] S. Zhang, Y. Zhan, and D. N. Metaxas, "Deformable segmentation via sparse representation and dictionary learning," *Medical Image Analysis*, vol. 16, no. 7, pp. 1385–1396, 2012.

[19] R. Sarkar, S. Das, and N. Vaswani, "Tracking sparse signal sequences from nonlinear/non-gaussian measurements and applications in illumination-motion tracking," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6615–6619.

[20] H. Li, C. Shen, and Q. Shi, "Real-time visual tracking using compressive sensing," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1305–1312.

[21] R. Sarkar, A. Vaccari, and S. T. Acton, "Sspared: Saliency and sparse code analysis for rare event detection in video," in *IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, July 2016, pp. 1–5.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[23] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[24] B. S. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 18, no. 8, pp. 837–842, 1996.

[25] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International journal of computer vision*, vol. 1, no. 4, pp. 321–331, 1988.

[26] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *Image Processing, IEEE Trans. on*, vol. 7, no. 3, pp. 359–369, 1998.

[27] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 24, no. 7, pp. 971–987, 2002.

[28] L.-K. Soh and C. Tsatsoulis, "Texture analysis of sar sea ice imagery using gray level co-occurrence matrices," *IEEE Transactions on geoscience and remote sensing*, vol. 37, no. 2, pp. 780–795, 1999.

[29] A. Jain and G. Healey, "A multiscale representation including opponent color features for texture recognition," *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 124–128, 1998.

[30] T. F. Chan, L. Vese *et al.*, "Active contours without edges," *Image processing, IEEE Trans. on*, vol. 10, no. 2, pp. 266–277, 2001.

[31] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2790–2797.

[32] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, Jan 2008.

[33] B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, June 2011, pp. 3313–3320.

[34] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 33, no. 11, pp. 2259–2272, 2011.

[35] R. Sarkar, S. Mukherjee, and S. T. Acton, "Shape descriptors based on compressed sensing with application to neuron matching," in *2013 Asilomar Conference on Signals, Systems and Computers*. IEEE, Nov 2013, pp. 970–974.

[36] H. Van Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Design of non-linear kernel dictionaries for object recognition," *Image Processing, IEEE Transactions on*, vol. 22, no. 12, pp. 5123–5135, 2013.

[37] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Low-rank sparse learning for robust visual tracking," in *European conference on computer vision*. Springer, 2012, pp. 470–484.

[38] D. Donoho, "Compressed sensing," *IEEE Trans. on Information Theory*, vol. 52(4), pp. 1289–1306, April 2006.

[39] E. Candes and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. on Information Theory*, vol. 52(12), pp. 5406 – 5425, December 2006.

[40] E. Candes, "Compressive sampling," in *Int. Congress of Mathematics, Madrid, Spain*, vol. 3, 2006, pp. 1433–1452.

[41] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, 2010.

[42] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.

[43] T.-W. Lee, "Independent component analysis," in *Independent Component Analysis*. Springer, 1998, pp. 27–66.

[44] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, vol. 5. IEEE, 1999, pp. 2443–2446.

[45] J. Tropp, A. C. Gilbert *et al.*, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Trans. on*, vol. 53, no. 12, pp. 4655–4666, 2007.

[46] Q. Qiu, V. M. Patel, and R. Chellappa, "Information-theoretic dictionary learning for image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2173–2184, 2014.

[47] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 543–550.

[48] S. Shekhar, V. M. Patel, H. V. Nguyen, and R. Chellappa, "Generalized domain-adaptive dictionaries," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 361–368.

[49] S. Bahrampour, N. M. Nasrabadi, A. Ray, and W. K. Jenkins, "Multimodal task-driven dictionary learning for image classification," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 24–38, 2016.

[50] A. Shrivastava, J. K. Pillai, V. M. Patel, and R. Chellappa, "Learning discriminative dictionaries with partially labeled data," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*. IEEE, 2012, pp. 3113–3116.

[51] M. Elad and M. Aharon, "Image denoising via learned dictionaries and sparse representation," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1. IEEE, 2006, pp. 895–900.

[52] B. Li and S. T. Acton, "Active contour external force using vector field convolution for image segmentation," *Image Processing, IEEE Transactions on*, vol. 16, no. 8, pp. 2096–2106, 2007.

[53] ——, "Automatic active model initialization via poisson inverse gradient," *Image Processing, IEEE Trans. on*, vol. 17, no. 8, pp. 1406–1420, 2008.

[54] S. Mukherjee and S. T. Acton, "Region based segmentation in presence of intensity inhomogeneity using legendre polynomials," *Signal Processing Letters, IEEE*, vol. 22, no. 3, pp. 298–302, 2015.

[55] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–79, 1997.

[56] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[57] C. Li, C.-Y. Kao, J. C. Gore, and Z. Ding, "Minimization of region-scalable fitting energy for image segmentation," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1940–1949, 2008.

[58] S. Lankton and A. Tannenbaum, "Localizing region-based active contours," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2029–2039, 2008.

[59] X.-F. Wang, D.-S. Huang, and H. Xu, "An efficient local chan–vese model for image segmentation," *Pattern Recognition*, vol. 43, no. 3, pp. 603–618, 2010.

[60] H. Feng, D. A. Castanon, and W. C. Karl, "Tomographic reconstruction using curve evolution," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2000, pp. 361–366.

[61] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the mumford and shah model," *International Journal of Computer Vision*, vol. 50, no. 3, pp. 271–293, 2002.

[62] M.-H. Cardinal, J. Meunier, G. Soulez, R. L. Maurice, É. Therasse, and G. Cloutier, "Intravascular ultrasound image segmentation: a three-dimensional fast-marching method based on gray level distributions," *IEEE Trans. Med. Imag.*, vol. 25, no. 5, pp. 590–601, 2006.

[63] J. G. Thomas, R. A. Peters, P. Jeanty *et al.*, "Automatic segmentation of ultrasound images using morphological operators," *IEEE Trans. Med. Imag.*, vol. 10, no. 2, pp. 180–186, 1991.

[64] J. A. Noble and D. Boukerroui, "Ultrasound image segmentation: a survey," *IEEE Trans. Med. Imag.*, vol. 25, no. 8, pp. 987–1010, 2006.

[65] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.* IEEE, 2010, pp. 3501–3508.

[66] D. Shen, Y. Zhan, and C. Davatzikos, "Segmentation of prostate boundaries from ultrasound images using statistical shape model," *IEEE Trans. Med. Imag.*, vol. 22, no. 4, pp. 539–551, 2003.

[67] M. Rousson and D. Cremers, "Efficient kernel density estimation of shape and intensity priors for level set segmentation," in *Intl. Conf. Medical Image Computing and Computer Assisted Intervention (MICCAI).* Springer, 2005, pp. 757–764.

[68] A. Tsai, A. Yezzi Jr, W. Wells, C. Tempany, D. Tucker, A. Fan, W. E. Grimson, and A. Willsky, "A shape-based approach to the segmentation of medical imagery using level sets," *IEEE Trans. Med. Imag.*, vol. 22, no. 2, pp. 137–154, 2003.

[69] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[70] T. Chan and W. Zhu, "Level set based shape prior segmentation," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2005, pp. 1164–1170.

[71] S. Roy, Q. He, E. Sweeney, A. Carass, D. S. Reich, J. L. Prince, and D. L. Pham, "Subject-specific sparse dictionary learning for atlas-based brain mri segmentation," *IEEE journal of biomedical and health informatics*, vol. 19, no. 5, pp. 1598–1609, 2015.

[72] X. He, R. S. Zemel, and M. Á. Carreira-Perpiñán, "Multiscale conditional random fields for image labeling," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).*

[73] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape," *International journal of computer vision*, vol. 72, no. 2, pp. 195–215, 2007.

[74] D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance," *International journal of Remote sensing*, vol. 28, no. 5, pp. 823–870, 2007.

[75] R. M. Haralick, K. Shanmugam *et al.*, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.

[76] A. Vailaya, M. A. Figueiredo, A. K. Jain, and H.-J. Zhang, "Image classification for content-based indexing," *IEEE transactions on image processing*, vol. 10, no. 1, pp. 117–130, 2001.

[77] A. Baraldi and F. Parmiggiani, "A neural network for unsupervised categorization of multivalued input patterns: an application to satellite image clustering," *IEEE Transactions on Geoscience and remote Sensing*, vol. 33, no. 2, pp. 305–316, 1995.

[78] Y. Chen, J. Z. Wang, and R. Krovetz, "Clue: cluster-based retrieval of images by unsupervised learning," *IEEE transactions on Image Processing*, vol. 14, no. 8, pp. 1187–1201, 2005.

[79] D. Dueck and B. J. Frey, "Non-metric affinity propagation for unsupervised image categorization," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.

[80] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995. [Online]. Available: http://dx.doi.org/10.1007/BF00994018

[81] E. A. Wan, "Neural network classification: A bayesian interpretation," *IEEE Transactions on Neural Networks*, vol. 1, no. 4, pp. 303–305, 1990.

[82] X. T. Yuan, X. Liu, and S. Yan, "Visual classification with multitask joint sparse representation," *IEEE Transactions on Image Processing*, vol. 21, no. 10, pp. 4349–4360, Oct 2012.

[83] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *CVPR 2011*, June 2011, pp. 625–632.

[84] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 3360–3367.

[85] S. Gao, I. W. H. Tsang, L. T. Chia, and P. Zhao, "Local features are not lonely-laplacian sparse coding for image classification," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2010, pp. 3555–3561.

[86] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1794–1801.

[87] J. Mairal, J. Ponce, G. Sapiro, A. Zisserman, and F. R. Bach, "Supervised dictionary learning," in *Advances in neural information processing systems*, 2009, pp. 1033–1040.

[88] A. Shrivastava, V. M. Patel, and R. Chellappa, "Multiple kernel learning for sparse representation-based classification," *Image Processing, IEEE Transactions on*, vol. 23, no. 7, pp. 3013–3024, 2014.

[89] L. Zhang, W.-D. Zhou, P.-C. Chang, J. Liu, Z. Yan, T. Wang, and F.-Z. Li, "Kernel sparse representation-based classifier," *Signal Processing, IEEE Transactions on*, vol. 60, no. 4, pp. 1684–1695, 2012.

[90] S. Gao, I. W. Tsang, and L.-T. Chia, "Sparse representation with kernels," *Image Processing, IEEE Transactions on*, vol. 22, no. 2, pp. 423–434, 2013.

[91] R. Sarkar, S. Ozer, K. Skadron, and S. Acton, "Image classification by multi-kernel dictionary learning," in *Signals, Systems and Computers, 2014 48th Asilomar Conference on*. IEEE, Nov 2014, pp. 73–77.

[92] M. Isard and A. Blake, "Condensationconditional density propagation for visual tracking," *International journal of computer vision*, vol. 29, no. 1, pp. 5–28, 1998.

[93] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[94] Y. Ke, R. Sukthankar, and M. Hebert, "Efficient visual event detection using volumetric features," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1. IEEE, 2005, pp. 166–173.

[95] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 3, pp. 555–560, 2008.

[96] J. Wu, J. M. Rehg, and M. D. Mullin, "Learning a rare event detection cascade by direct feature selection." in *NIPS*, vol. 4, 2003, pp. 855–861.

[97] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: Multitarget detection and tracking," in *European Conference on Computer Vision*. Springer, 2004, pp. 28–39.

[98] R. E. Kalman *et al.*, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[99] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2042–2049.

[100] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank, "A system for learning statistical motion patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1450–1464, Sept 2006.

[101] A. Basharat, A. Gritai, and M. Shah, "Learning object motion patterns for anomaly detection and improved object detection," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[102] A. Ravichandran and S. Soatto, "Long-range spatio-temporal modeling of video with application to fire detection," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 329–342.

[103] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–819.

[104] F. F. E. Guraya, F. A. Cheikh, A. Tremeau, Y. Tong, and H. Konik, "Predictive saliency maps for surveillance videos," in *Distributed Computing and Applications to Business Engineering and Science (DCABES), 2010 Ninth International Symposium on*. IEEE, 2010, pp. 508–513.

[105] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," 2010.

[106] O. Boiman and M. Irani, "Detecting irregularities in images and in video," *International Journal of Computer Vision*, vol. 74, no. 1, pp. 17–31, 2007.

[107] J. Kwon and K. M. Lee, "A unified framework for event summarization and rare event detection," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012, pp. 1266–1273.

[108] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *CVPR 2011*, June 2011, pp. 3449–3456.

[109] ——, "Abnormal event detection in crowded scenes using sparse representation," *Pattern Recognition*, vol. 46, no. 7, pp. 1851–1864, 2013.

[110] T. Bouwmans, A. Sobral, S. Javed, S. K. Jung, and E.-H. Zahzah, "Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset," *Computer Science Review*, 2016.

[111] X. Zhou, C. Yang, and W. Yu, "Moving object detection by detecting contiguous outliers in the low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 597–610, 2013.

[112] X. Cui, J. Huang, S. Zhang, and D. N. Metaxas, "Background subtraction using low rank and group sparsity constraints," in *European Conference on Computer Vision*. Springer, 2012, pp. 612–625.

[113] X. Liu, G. Zhao, J. Yao, and C. Qi, "Background subtraction based on low-rank and structured sparse decomposition," *IEEE Transactions on Image Processing*, vol. 24, no. 8, pp. 2502–2514, 2015.

[114] J. Wen, Y. Xu, J. Tang, Y. Zhan, Z. Lai, and X. Guo, "Joint video frame set division and low-rank decomposition for background subtraction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 12, pp. 2034–2048, 2014.

[115] F. Zhang, J. Yang, Y. Tai, and J. Tang, "Double nuclear norm-based matrix decomposition for occluded image recovery and background modeling," *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1956–1966, 2015.

[116] N. Kwak and C.-H. Choi, "Input feature selection by mutual information based on parzen window," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 24, no. 12, pp. 1667–1671, 2002.

[117] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 27, no. 8, pp. 1226–1238, 2005.

[118] F. Fleuret, "Fast binary feature selection with conditional mutual information," *The Journal of Machine Learning Research*, vol. 5, pp. 1531–1555, 2004.

[119] B. Epshtein and S. Uliman, "Feature hierarchies for object classification," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1. IEEE, 2005, pp. 220–227.

[120] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[121] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.

[122] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Computer vision and pattern recognition, 2009. cvpr 2009. ieee conference on*. IEEE, 2009, pp. 1597–1604.

[123] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Advances in neural information processing systems*, 2006, pp. 545–552.

[124] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature reviews neuroscience*, vol. 2, no. 3, pp. 194–203, 2001.

[125] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 11, pp. 1254–1259, 1998.

[126] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 733–740.

[127] W. Wang, Y. Song, and A. Zhang, "Semantics-based image retrieval by region saliency," in *Image and Video Retrieval*. Springer, 2002, pp. 29–37.

[128] A. Papushoy and A. G. Bors, "Image retrieval based on query by saliency content," *Digital Signal Processing*, vol. 36, pp. 156–173, 2015.

[129] H. Li and K. N. Ngan, "Saliency model-based face segmentation and tracking in head-and-shoulder video sequences," 2008.

[130] V. Mahadevan and N. Vasconcelos, "Saliency-based discriminant tracking."

[131] C. Galleguillos, B. McFee, S. Belongie, and G. Lanckriet, "Multi-class object localization by combining local contextual interactions," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 113–120.

[132] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 2, pp. 353–367, 2011.

[133] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012, pp. 853–860.

[134] O. Le Meur, P. Le Callet, D. Barba, and D. Thoreau, "A coherent computational approach to model bottom-up visual attention," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 5, pp. 802–817, 2006.

[135] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 3166–3173.

[136] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *Image Processing, IEEE Trans. on*, vol. 13, no. 4, pp. 600–612, 2004.

[137] J. Goldberger, S. Gordon, and H. Greenspan, "An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures," in *Computer Vision, 2003. Proc. Ninth IEEE International Conference on.* IEEE, 2003, pp. 487–493.

[138] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[139] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 2169–2178.

[140] H. Ling and K. Okada, "Diffusion distance for histogram comparison," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1. IEEE, 2006, pp. 246–253.

[141] ——, "An efficient earth mover's distance algorithm for robust histogram comparison," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 840–853, May 2007.

[142] P.-P. Vázquez and J. Marco, "Using normalized compression distance for image similarity measurement: an experimental study," *The Visual Computer*, vol. 28, no. 11, pp. 1063–1084, 2012.

[143] B. J. Campana and E. J. Keogh, "A compression-based distance measure for texture," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 3, no. 6, pp. 381–398, 2010.

[144] C. H. Bennett, P. Gács, M. Li, P. M. Vitányi, and W. H. Zurek, "Information distance," *Information Theory, IEEE Trans. on*, vol. 44, no. 4, pp. 1407–1423, 1998.

[145] P. M. Vitányi, F. J. Balbach, R. L. Cilibrasi, and M. Li, "Normalized information distance," in *Information theory and statistical learning.* Springer, 2009, pp. 45–82.

[146] T. Watanabe, K. Sugawara, and H. Sugihara, "A new pattern representation scheme using data compression," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 24, no. 5, pp. 579–590, May 2002.

[147] A. J. Pinho and P. J. Ferreira, "Image similarity using the normalized compression distance based on finite context models," in *Image Processing (ICIP), 2011 18th IEEE International Conference on.* IEEE, 2011, pp. 1993–1996.

[148] R. Sarkar and S. T. Acton, "Sdl: saliency based dictionary learning for image similarity," in *2016 IEEE Trans. on Image Processing (under review).*

[149] G. Xiao, M. Brady, J. A. Noble, and Y. Zhang, "Segmentation of ultrasound b-mode images with intensity inhomogeneity correction," *IEEE Trans. Med. Imag.*, vol. 21, no. 1, pp. 48–57, 2002.

[150] C. Li, R. Huang, Z. Ding, J. Gatenby, D. N. Metaxas, and J. C. Gore, "A level set method for image segmentation in the presence of intensity inhomogeneities with application to mri," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 2007–2016, 2011.

[151] N. Barth, "The gramian and k-volume in n-space: some classical results in linear algebra," *J Young Investig*, vol. 2, 1999.

[152] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 34, no. 11, pp. 2274–2282, 2012.

[153] A. Jinda-Apiraksa, V. Vonikakis, and S. Winkler, "California-nd: An annotated dataset for near-duplicate detection in personal photo collections," in *Quality of Multimedia Experience (QoMEX), 2013 Fifth International Workshop on.* IEEE, 2013, pp. 142–147.

[154] R. Cilibrasi and P. Vitanyi, "Clustering by compression," *Information Theory, IEEE Trans. on*, vol. 51, no. 4, pp. 1523–1545, 2005.

[155] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *Information Theory, IEEE Trans. on*, vol. 44, no. 6, pp. 2743–2760, 1998.

[156] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[157] J. N. Kather, C.-A. Weis, F. Bianconi, S. M. Melchers, L. R. Schad, T. Gaiser, A. Marx, and F. G. Zöllner, "Multi-class texture analysis in colorectal cancer histology," *Scientific Reports*, vol. 6, 2016.

[158] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE transactions on pattern analysis and machine intelligence*, vol. 12, no. 1, pp. 55–73, 1990.

[159] L. E. Boucheron, B. Manjunath, and N. R. Harvey, "Use of imperfectly segmented nuclei in the classification of histopathology images of breast cancer," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2010, pp. 666–669.

[160] J. N. Kather, F. G. Zllner, F. Bianconi, S. M. Melchers, L. R. Schad, T. Gaiser, A. Marx, and C.-A. Weis, "Collection of textures in colorectal cancer histology," May 2016. [Online]. Available: https://doi.org/10.5281/zenodo.53169

[161] Y.-M. Kim, S. W. Choi, and S.-W. Lee, "Fast scene change detection using direct feature extraction from mpeg compressed videos," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, vol. 3. IEEE, 2000, pp. 174–177.

[162] Y. Zhou, S. Yan, and T. S. Huang, "Detecting anomaly in videos from trajectory similarity analysis," in *Multimedia and Expo, 2007 IEEE International Conference on*, July 2007, pp. 1087–1090.

[163] Y. Ke, R. Sukthankar, and M. Hebert, "Event detection in crowded videos," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.

[164] V. Mahadevan and N. Vasconcelos, "Background subtraction in highly dynamic scenes," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–6.

[165] Y. Peng, J. Suo, Q. Dai, and W. Xu, "Reweighted low-rank matrix recovery and its application in image restoration," *IEEE transactions on cybernetics*, vol. 44, no. 12, pp. 2418–2430, 2014.

[166] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *Pattern Analysis and Machine Intelligence, IEEE Trans. on*, vol. 34, no. 10, pp. 1915–1926, 2012.

[167] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[168] Y. Yankelevsky and M. Elad, "Dual graph regularized dictionary learning," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 611–624, 2016.

[169] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *Journal of machine learning research*, vol. 11, no. Aug, pp. 2287–2322, 2010.

[170] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[171] H. Van Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Kernel dictionary learning," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2021–2024.

[172] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 221–228.

[173] H. Hino, N. Reyhani, and N. Murata, "Multiple kernel learning by conditional entropy minimization," in *Machine Learning and Applications (ICMLA), 2010 Ninth International Conference on*. IEEE, 2010, pp. 223–228.