Discriminative Dictionary Learning by Exploiting Inter-Class Similarity for HEp-2 Cell Classification

Discriminative Dictionary Learning by Exploiting Inter-Class Similarity for HEp-2 Cell Classification

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology
in
Computer Science

by

Aditya Panda
Roll No: CS1723

under the guidance of

Prof. Pradipta Maji
Professor
Machine Intelligence Unit

Indian Statistical Institute
Kolkata-700108, India

July 2019

CERTIFICATE

This is to certify that the dissertation entitled **"Discriminative Dictionary Learning by Exploiting Inter-Class Similarities for HEp-2 Cell Classification"** submitted by **Aditya Panda** to Indian Statistical Institute, Kolkata, in partial fulfillment for the award of the degree of **Master of Technology in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

Pradipta Maji
Professor,
Machine Intelligence Unit,
Indian Statistical Institute,
Kolkata-700108, INDIA.

Acknowledgments

I would like to convey my highest gratitude to my advisor, Prof. Pradipta Maji, Machine Intelligence Unit, Indian Statistical Institute, Kolkata, for his patience and encouragement, during preparation of this thesis. I am indebted to all my teachers at Indian Statistical Institute for giving me the insight into computer science, through their teaching. Finally, I am very much grateful to my parents for their everlasting support.

Aditya Panda
Indian Statistical Institute
Kolkata - 700108 , India.

Abstract

In this literature we present an algorithm for automatic classification of IIF images of HEp-2 cells into relevant classes. Our algorithm is majorly based on the "Dictionary Learning" algorithm and we have redefined it's objective function to suit our purpose. The major difficulty in HEp-2 cell image classification lies in it's low inter-class variability and substantial intra-class variations. To address these issues, we have modified the objective function of "Dictionary Learning" to learn inter-class features. Moreover, we used a local feature extractor based pre-processing stage and also a "spatial decomposition" classifier set-up for better classifying test images. We evaluated our algorithm on three most widely accepted bamechmark data-sets for HEp-2 cell classification, ICPR 2012, ICIP 2013 and SNP data-sets. Proposed algorithm has achieved superior results than other popular dictionary learning algorithms for HEp-2 cell classification. Moreover, when comparing with other algorithms for HEp-2 cell classification, including the winners of ICPR 2012, ICIP 2013 and SNP data-set, we show that proposed algorithm reports very competitive result. Though our proposed algorithm is designed to be application specific to HEp-2 cell, still we evaluated its performance on another popular benchmark data-set, "Diabetic Retinopathy" data-set. Our algorithm provided higher accuarcy than other state-of-the-art algorithms on that data-set too.

Keywords: *Dictionary Learning, Indirect Immuno-fluorescence Image,Cell Classification, Human Epithelial Cell-2, Diabetic Retinopathy*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The circulatory system in the human body transports micro-particles to facilitate a wide spectrum of functions. The immunity system defends our body by detecting foreign pathogens and attacking the invasions. Immunity in human mainly work through two pathways, internal and externally initiated processes. The body's inherent self-defence mechanism comprises of native micro-organisms, which counters pathogens, without presence of any external aid. On the other hand, humans also acquire the ability to defend against pathogens, as the body learns to counter infections and develops antibodies against the pathogens.

This acquired form of immunity, is sometimes an imperfect process and might occasionally learn to incorrectly identify our body's own cells as germ and generate antibodies to defend against these native cells. Such situations are identified as "auto-immune diseases". These antibodies specifically attack body cell's nucleus. So they are termed Antinuclear Auto-antibodies (ANAs). This produces some common illness and are characterized by a chronic inflammation in different organs.

The common tests used for detecting and quantifying ANAs are indirect Immune-Fluorescence (IIF) and Enzyme Linked Immunosorbent Assay (ELISA) tests. The IIF test is preferred and recommended as the ELISA test has limited detection application scope [28].

HEp-2 cells are available at inner cell linings of human larynx. It bonds with serum antibody forming a molecular complex. This complex then reacts with human immunoglobulin, and bonds with added fluoro-chrome. Fluoro-chrome makes it visible under a fluorescence microscope. This image, when observed under microscope reveals the antigen-antibody patterns. Medical experts examine the images and classify the staining patterns for each cell into different classes of interest. The computer aided automated recognition of these classes is a "pattern recognition" problem and is the key to an efficient and automatic diagnosis of patients with these ailment. For a more detailed description the readers can go through [14].

However, the manual classification of cells using IIF method suffers from few drawbacks. The major disadvantages are: the low level of standardization, the inter-observer variability, which reduces reproducibility of reports. Also there is lack of resources and experinced physicians. Another problem reported is the similarity between some classes, which causes the interpretational errors. The computer aided automatic classification of HEp-2 cells can pave the way for more elegant solution to these problems.

In recent years, many researchers tried with different approaches for HEp 2 cell classification. In this article, we have proposed to classify the HEp-2 cell images using "Dictionary Learning". However, as already mentioned, major difficulty in HEp-2 image classification is due to a low inter-class variation and also a substantial intra-class variation. Moreover, number of patients is not same from all the classes of ailment. So in many cases, biomedical image data set,including this HEp-2 data-set, suffers from class imbalance. To circumvent these issues, we have modified the objective function of "Dictionary Learning" to incorporate inter-class dependency. Moreover, we used a local feature extractor based pre-processing stage and also a SVM based "spatial decomposition" classifier to classify the test image. We evaluate our algorithm on the ICPR 2012, ICIP 2013 and SNP competition data sets. The results have been compared with other state-of-the-art algorithms. We also evaluated our algorithms performance on another classification problem, detection of diabetic retionpathy by classification of fundus images.

The remaining part of the article has been arranged in the following sections. Section 2 reviews the major dictionary learning algorithm as well as different approaches to classify HEp-2 cells. Section 3 discusses in depth about our proposed algorithm. The next section 4 discuses about the performance of our proposed algorithm and compares it with other state of the art algorithm on HEp-2 data-sets and diabetic retinopathy data-sets. Finally Section 5 summarizes the algorithm and discusses the future scope of work.

Chapter 2

Brief Literature Review

2.1 Dictionary Learning

"Dictionary Learning" algorithms try to learn "features" from the training data-set, such that, any new signal generated from the same distribution as that of the training signal source, can be expressed as a linear combination of a few "learned features". The collection of "learned features" is called the "Dictionary". Feature vectors contained in the dictionary are also called "atoms". In the current literature we use the words "features" and "atoms" interchangeably.

Let Y be d dimensional signal, $Y \in \mathbb{R}^d$. D be the dictionary with K features and each feature has dimension of d (same as that of the signal), $D \in \mathbb{R}^{d \times K}$. Here, we try to express the signal Y to be linear combination of the atoms from dictionary,

$$Y = DX \tag{2.1}$$

where X is coefficient matrix, $X \in \mathbb{R}^K$. X contents the information about which of the features (from K features in the dictionary) are used for a particular signal representation. In figure 2.1, the red cells in the sparse representation vector, X, represents the index of features in the dictionary which are used for reconstruction of the signal. Dictionary learning wants the signal to be represented as linear combination of fewest features possible. Hence during the training the X vector is modelled to be, as sparse as possible.

"Sparse Representation" system of a signal, has shown strong relationship or similarity to human vision system. The human vision system is highly selective to some specific common features like shape, color etc. Similarly the sparse systems try to represent each signal as a linear combination of a few dictionary features. In [34] [35], it is suggested that sparse visual structures has close similarity with working of V1 sector of primary visual cortex. This similarity and applications based on it, has inspired many researchers in recent years in this field. Sparse representation based Dictionary learning algorithms have reported competitive results in signal restoration

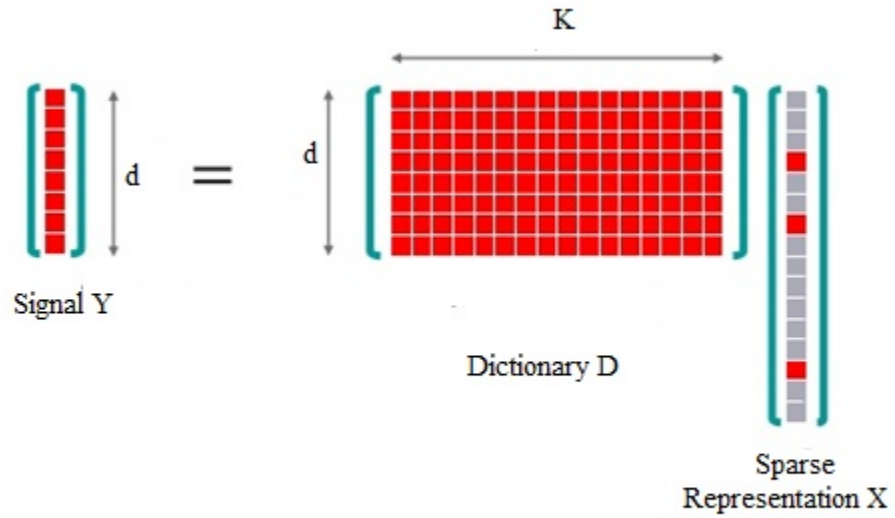

Figure 2.1: Signal representation as a linear combination of features from dictionary

[10],image compression [6], image super resolution [51], object recognition [52] etc.

To induce sparsity in signal representation, "Dictionary Learning" algorithm uses an over-complete dictionary. Over-complete-ness suggests that there should be more dictionary atoms than number of dimensions in the signal. In reference to figure 2.1 above, we have for and over-complete systems $K > d$. However to represent a d dimensional signal by using more than d dimension will have redundancy and the equation $Y = DX$ will have infinite number of possible combination of atoms to represent a signal. In other words for a known D and known Y there will be infinite number of possible X matrix and we select that X which is sparsest or has least number of non-zero elements.
The objective function can be formulated as:

$$< D, X >= \underset{D,X}{\operatorname{argmin}} \|X\|_0 such\ \ that\ \ Y = DX \tag{2.2}$$

or in constrained form and specified sparsity limit:

$$< D, X >= \underset{D,X}{\operatorname{argmin}} \|Y - DX\|_F^2 + \lambda \|X\|_0 \tag{2.3}$$

The l_0 norm induces sparsity and is defined as number of non zero coefficients in the argument vector or matrix. The frobenius Norm is basically square root of sum of square of the elements of the matrix

$$\|A_{M\times N}\|_F^2 = \sum_{i=1}^{M}\sum_{j=1}^{N}\sqrt{\|a_{ij}\|^2} \tag{2.4}$$

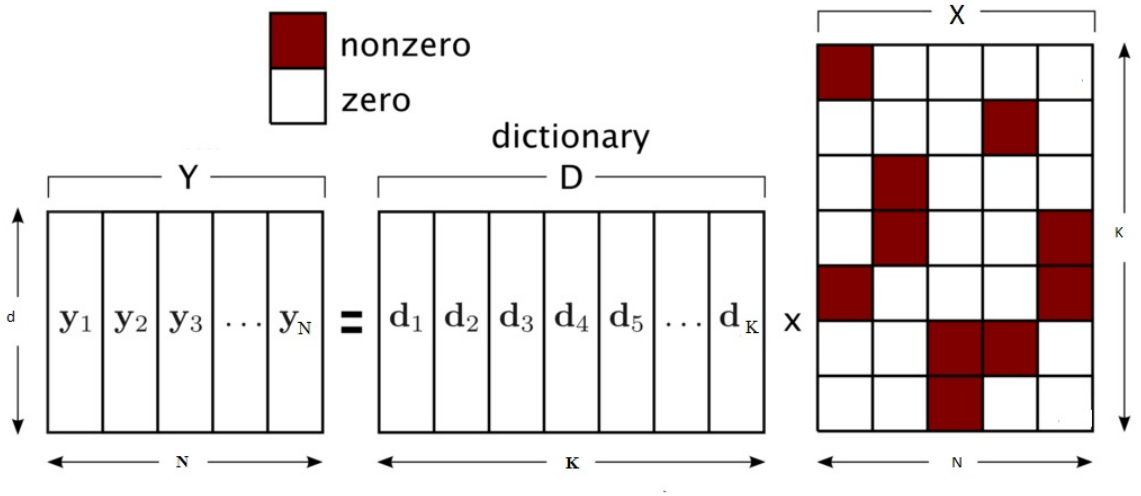

Figure 2.2: Schematic diagram for training dictionary on N number of signals. Training signals are stacked one after another, to form the signal matrix

However, finding the optimal solution to $Y = DX$ while finding the sparsest X matrix is exponential of computational cost. It can be solved using greedy approach. Also some convex relaxation approaches exists. In 2009, Wright et. al. [50] had shown that, if the solution is sufficiently sparse then l_0 norm can be approximated by l_1 norm.

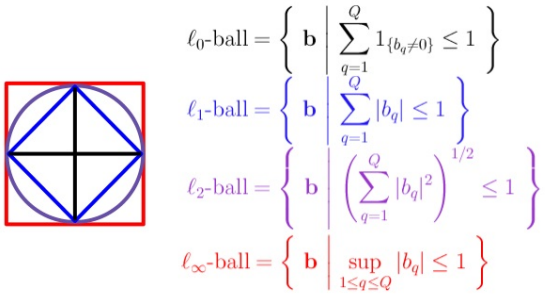

Figure 2.3: Different orders of norm

So the objective function is replaced as:

$$< D, X >= \underset{D,X}{\operatorname{argmin}} \|Y - DX\|_F^2 + \lambda\|X\|_1 \tag{2.5}$$

2.2 Review of Relevant Dictionary Learning Algorithms

2.2.1 K-SVD Algorithm

K-SVD proposed by Aharon et.al. [1] in 2006, is one of the most popular algorithms for dictionary learning. In the K-SVD algorithm, for a given signal, the dictionary (D) is first initialized and the best coefficient matrix X is found. After finding X, the algorithm searches for a better dictionary D. This completes one iteration. This process is repeated several times until threshold number of iteration achieved or the desired accuracy is reached.

However, finding the whole dictionary, at once requires complex analysis. So one atom of the dictionary (D) updated, at a time, while keeping X fixed. The k^{th} atom is updated by making it as perfect as possible by reducing the error caused by that atom. Singular Value Decomposition is used to solve the equation. The algorithm is named K-SVD to mimic it's similarity to K-Means algorithm. K-Means tries to cluster around K centers. Similarly, K-SVD considers dictionary atoms as cluster heads in d dimensional space and tries to cluster around those dictionary atoms.

Finding the truly optimal X, is of exponential complexity. However, the authors used approximation pursuit method. Any pursuit algorithm such as, the Orthogonal Matching Pursuit (OMP) [7] can be used for the calculation of the sparse coefficient matrix X. Next, we formally state the Orthogonal Matching Pursuit algorithm and the complete K-SVD algorithm pseudo-code. We have stated K-SVD algorithm in detail, because it has been used in in our proposed algorithm.

Algorithm 1 Orthogonal Matching Pursuit Algorithm

Input: Dictionary D and the input signal Y
Output: Sparse representation X
$t \leftarrow 1$
$R_t \leftarrow Y$
while $t \leq MAX\_ITER$ **do**
  find atom d_j which has maximum inner product $|\langle R_t, d_j \rangle|$
  $X_j \leftarrow \frac{\langle R_t, d_j \rangle}{\|d_j\|}$
  $R_{t+1} \leftarrow R_t - X_j d_j$
  $t \leftarrow t + 1$
end

Here X_j denote the j^{th} row of the sparsity matrix (X) and d_j denotes the j^{th} atom of the dictionary(D).

The complete K-SVD algorithm incorporating the dictionary update stage and the sparse coding stage given as:

Algorithm 2 The KSVD Algorithm

Input: Sparse representation X and the input signal Y
Output: Dictionary D
Set the initial dictionary matrix D with l_2 normalized columns or atoms
while *convergence not reached* **do**
Sparse Coding stage: use any pursuit algorithm (In our case it is OMP) to get the coefficient matrix X_i for each signal Y_i

Dictionary Update Stage:
for k in range number Of Atoms
Define the group of signals that use this atom $\omega_k = (i|X_k^i \neq 0)$
Compute the overall reconstruction error matrix $E_k = Y - \sum_{j \neq k} d_j X_j^T$
Restrict E_k by choosing columns of E_K corresponding to ω_k and obtain E_K^R
Apply SVD decomposition to get $E_K^R = U \Delta V^T$ The updated K_{th} atom $\tilde{d_k}$ is first
column of U and X_k^R is $\Delta[1,1]$ times first column of V
end

2.2.2 Other relevant algorithms

The major problem with K SVD was, it was not suitable for classification purpose. All the classes were using same dictionary or same set of features. However since K-SVD was proposed, many researchers have come up with many solutions to image classification using dictionary learning. Most of them try to remodel the dictionary and considered one sub-dictionary for each class, to capture class-specific features. In 2010, Ramirez et. al. [42] utilised the idea of class specific sub-dictionary and proposed class Dictionary Learning With Structured Incoherence (DLSI). They considered the objective function as class specific reconstruction error. The objective function is:

$$\langle D_i, X_i \rangle = \underset{D_i, X_i}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\|Y_i - D_i X_i\|_F^2 + \lambda_1 \sum_{j=1, j \neq i}^{\mathbb{C}} \left\| D_i^T D_j \right\|_F^2 + \lambda_2 \|X_i\|_1 \right) \tag{2.6}$$

The reconstruction error is used for classification of test signal. The classification scheme can be formulated as

$$\hat{c} = \underset{i \in (1,2,3 \ \ldots \ \mathbb{C})}{\operatorname{argmin}} \left(\|Y_i - D X_i\|_F^2 + \lambda_1 \sum_{j=1, j \neq i}^{\mathbb{C}} \left\| D_i^T D_j \right\|_F^2 + \lambda_2 \|X_i\|_1 \right) \Bigg) \tag{2.7}$$

where, $D = [D_1 \ D_2 \ ... \ D_{\mathbb{C}}]$ is the dictionary, $D_i \in \mathbb{R}^{d \times K_i}$ is the sub dictionary of i^{th} class and K_i is number of atom is i^{th} class specific dictionary. Y_i is the set of signals with label of i^{th} class. $\mathbb{C}$ is total number of class in the training data and X_i is the sparse representation of the i_{th} class specific signal Y_i corresponding to complete dictionary D.

However soon after the success of the DLSI, Kong and Wang [19] proposed Dictionary Learning With Commonality and Particularity (DL-COPAR). As a major improvement to previous approaches, they considered another separate sub-dictionary to store features common to all the classes. The relevant dictionary is called the commonality dictionary (D_0). The objective function as reported,

$$\langle D_i, X_i \rangle = \underset{D_i, X_i}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\|Y_i - DX_i\|_F^2 + \lambda_1 \|Y_i - D_i X_i - D_0 X_0\|_F^2 + \lambda_2 \sum_{j=1, j \neq i}^{\mathbb{C}} \left\|D_i^T D_j\right\|_F^2 + \lambda_3 \|X_i\|_1 \right) \tag{2.8}$$

the term $\sum_{j=1, j \neq i}^{\mathbb{C}} \left\|D_i^T D_j\right\|_F^2$ is added to make the sub dictionaries more disriminative. For classification they used similar reconstruction based classifier as in DLSI [42].

After DL-COPAR reported competitive results Yang et al. [53] proposed Fisher's Discriminative Dictionary Learning (FDDL) Algorithm which further imposed some restrictions on the sparse coefficient matrix, X. They expect for all the signals, with same class label, their sparse representation, must cluster as close as possible. Similarly sparse representation for signals from different classes should be as further clustered as possible. This follows from the intuition that signals from same class uses similar group of atoms, for reconstruction. For all signals Y_i in a class i let X_i be the corresponding sparse representation. The authors define the mean operation of matrix as $\mathcal{M} : A_{P \times Q} \rightarrow B_{P \times Q}$. The mean is taken over all the columns and one single column is prepared. However, to preserve the dimensions, that column vector is repeated and stacked Q times to get a single matrix of same dimension of input matrix. We have the mean of the i^{th} sparse coefficient matrix X_i is given as $M_i = \mathcal{M}(X_i)$ and $M = \mathcal{M}(\mathcal{M}_\setminus)$ The corresponding objective function is:

$$\langle D_i \rangle = \underset{D_i, X_i}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\|Y_i - DX_i\|_F^2 + \lambda_1 \sum_{j=1, j \neq i}^{\mathbb{C}} \left(\|D_i X_j\|_F^2 + \lambda_3 \|X_i\|_1 + R(X_i) \right) \right) \tag{2.9}$$

Where $R(X_i)$ is the main contribution of their algorithm and is given as

$$R(X_i) = \|X_i - M_i\|_F^2 - \|M_i - M\|_F^2 + \|X\|_F^2 \tag{2.10}$$

Here, the term $\|X\|_F^2$ is added for convex relaxation of the $R(X_i)$.

Kong et al [20] proposed an extension of K-SVD [1] algorithm for specific application of HEp 2 cell classification. It also considers sub dictionary for each class. For each atom in the dictionary, they tried to maximize it's reconstruction power for that particular class in which the atom is in. For the rest of the classes, it tries to decrease it's reconstruction power. This leads to highly discriminative dictionaries. Thus, overall the objective function is (let us update the k^{th} atom and let it's class label be c_k and all other remaining classes be denoted by $\overline{c}_k$)

$$\langle d_k \rangle = \underset{d_k}{\operatorname{argmin}} \left\| Y_{c_k} - \sum_{f(d_j)=c_k} d_j (x_j^T)_{c_k} - d_k (x_k^T)_{c_k} \right\|_F^2 - \left\| Y_{\overline{c_k}} - \sum_{f(d_j)=\overline{c_k}} d_j (x_j^T)_{\overline{c_k}} - d_k (x_k^T)_{\overline{c_k}} \right\|_F^2 \tag{2.11}$$

where, $f(d_k) = c_k$ i.e. f function extracts the labelled class for each atom in the dictionary and $\overline{c_k}$.They solved the optimization using the same Singular Value Decomposition method, as proposed in the original K-SVD literature [1]. The two terms in the objective function were solved separately. Since they oppose each other, the authors suggested to the final solution should be along the first solution and orthogonal to the second solution vector.

Recently, Low Rank Shared Dictionary Learning[47] was proposed by Monga et. al. Their framework closely follows the work done in FDDL [53] and they further proposed that the commonality dictionary part should necessarily have low rank. If commonality dictionary D_0 does not have low rank, then during the training the it may even absorb some class specific features. So in the objective function they used another term, nuclear norm of D_0. Nuclear norm is evaluated as sum of singular values of the argument matrix. The objective function is given as:

$$\begin{aligned} \langle D_i, X_i \rangle = \underset{D_i, X_i}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \Bigg(& \|Y_i - DX_i\|_F^2 + \lambda_1 \|Y_i - D_i X_i - D_0 X_0\|_F^2 + \lambda_2 \sum_{j=1, j\neq i}^{\mathbb{C}} \|D_i D_j\|_F^2 \\ & + \lambda_3 \|D_0\|_* + \lambda_4 \|X_i\|_1 + \|X_i - M_i\|_F^2 - \|M_i - M\|_F^2 + \|X\|_F^2 \Bigg) \end{aligned} \tag{2.12}$$

where $\|D_0\|_*$ is the nuclear norm of the commonality dictionary D_0

2.3 Relevant works on HEp-2 cell

In this section, we briefly review relevant algorithms on computer aided classification of HEp-2 cell. Roughly, we divide the existing methods into three categories, "clas-

sification using texture ",“classification using shape” and “classification using both texture and shape”.

- The “Texture-Based classification” approach majorly includes the Local Binary Patterns(LBP) based approaches [31],[32] and it's variants [2],[55],[38],[36]. LBPs are the widely used approaches to capture texture feature. Among local binary pattern-based algorithms reported in recent years, Co-occurrence of Adjacent Local Binary Pattern (CoALBP) [29], [30], Gradient-oriented Co-Occurrence of Local Binary Pattern (GoC-LBPs) [45], and Pairwise Rotation-Invariant Co-Occurrence of Local Binary Pattern (PRICoLBP) [39] were the three most successful algorithms. In [29], Nosaka and Fukui proposed to use CoALBP for the HEp-2 cell classification and performed the best in the contest for HEp-2 cell classication, which was held with the International Conference on Pattern Recognition (ICPR) 2012. In this approach, each image was filtered by a Gaussian function to remove noise and manually rotated with nine orientations (to improve the robustness to rotation), CoALBP features were extracted for all images (both the original images and the manually rotated images), and a linear support vector machine (SVM) was adopted for classication. In addition to the methods mentioned above, the original LBP [33], completed LBP [16], and other well-known texture features, e.g., maximum response filter banks (e.g., MR8) [46], gray-level co-occurrence matrices [17], and Wavelet [15], have also been used in the HEp-2 cell classication.

- In “Shape-Based Classification” approaches, researchers have tried to classify the images based on shapes of cells. They however extensively used the sell segmentation masks provided by the organizers of different competition. In [37], Ponomarev *et al.* exploited shape feature by counting the distribution of the number of objects of interest,(post segmentation) area of those segmented objects amongst other important features. However, though provided high classification accuracy due to its high sensitivity to mild noise in shape features, it is not widely used in practice. In [21], Larsen *et al.* introduced a novel second-order donut-like shape index histogram descriptor and was closely third winner of the HEp-2 cell classication contest held at the International Conference on Image Processing (ICIP) in 2013.

- We now briefly summarize the approaches in “Classification using Both Texture and Shape”. In [18], Kong et al. adopted Varma's MR8 method [46] to extract the texture features. For extracting the shape based features, they used Bag of Words approach for creating a vocabulary of shape based features. Finally, pyramid of Histogram of Oriented Gradients (HoG) [5] twas also used during the classification step. The texture and shape histogram were weighted and concatenated to create the final signal vector. In [41], Shen et al. proposed to combine PRICoLBP [39] and Bag of Words, with SIFT feature [24] for the

HEp-2 cell classication. The two sources of features were stacked one after another using a linear kernel support vector machine. In [27], Manivannan *et. al.* proposed a method based on combination of four different features and reported best accuracy in ICPR 2014 competition. In their method, each image response was taken in four orientations, multi-scale patches were sampled densely, four types of features were extracted. In total, sixteen histograms were obtained to train sixteen support vector machines with linear kernel. In addition, Theodorakopoulos *et. al.* also investigated the combination of different features, e.g., combining GoC-LBPs [45] and a multivariate distribution of SIFT features [44], combining the morphological features and a bundle of local gradient descriptors.

Chapter 3

Discriminative Dictionary Learning by Exploiting Inter-class Dependencies

3.1 Objective Function

One major difficulty for effective classification of HEp-2 cell lies in it's high inter-class similarity and high intra-class variability. The existing algorithms had specific measures to make the class specific dictionaries discriminative. However existing algorithms tend to make the sub-dictionaries discriminative. In other words they tend to reduce the overlap between the sub-space spanned by the atoms of the class specific dictionaries. However the inter-class similarity indicates that there is some overlapping region between sub-spaces spanned by different class specific dictionaries. So the existing dictionary learning based algorithms were not suitable for HEp-2 cell image classification.

So to address this issue, we proposed to modify the objective function to better capture the features common between the classes. The commonality dictionary can only captures the features which are common amongst all the classes such as the background etc. However there may be some features which are common between two or three classes and those features can not be captured by the commonality dictionary. While existing models which try to make the class specific dictionary excessive discriminative, they simply lose those between class features. So to better discover inter class features, we add clustering between class specific dictionaries and term it as "Family Specific Sub-dictionary". Each family comprises of a few classes. The resulting dictionary that we have considered is

$$D = \Big[\; D_0 \underbrace{D_1 \;\; D_2 \;\; D_3 \;\; ... \;\; D_{\mathbb{C}+i} \;\; ... \;\; D_{\mathbb{C}}}_{class \;\; specific \;\; dictionary} \; \underbrace{D_{\mathbb{C}+1} \;\; D_{\mathbb{C}+2} \;\; ... \;\; D_{\mathbb{C}+f} \;\; ... D_{\mathbb{C}+F}}_{family \;\; specific \;\; dictionary} \Big]$$

where $\mathbb{C}$ is the number of classes in the data-set and $\mathbb{F}$ is number of families in the data set. A more detailed discussion each section of the dictionaries are as follows

1. Class specific dictionary : Each class has a specific dictionary and it explores and stores the features specific to the that particular class. The i^{th} class specific dictionary is given as
$$D_i \in \mathbb{R}^{d \times K_i} \quad i = 1, 2, \ \ldots \ \mathbb{C}$$
where K_i is the number of atoms allowed to the i^{th} class. So the class specific dictionary component is
$$D_{class \ \ specific} = \begin{bmatrix} D_1 & D_2 & D_3 & \ldots & D_{C+i} & \ldots & D_C \end{bmatrix}$$

2. Family Specific Dictionary : A brief description of the classes of HEp-2 cell are as follows. This description in the following section from the website of the ICPR 2012 competition `https://mivia.unisa.it/datasets/biomedical-image-datasets/hep2-image-dataset/`

   (a) Homogeneous: diffuse staining of the inter-phase nuclei and staining of the chromatin of mitotic cells;

   (b) Fine speckled: fine granular nuclear staining of inter-phase cell nuclei;

   (c) Coarse speckled: coarse granular nuclear staining of inter-phase cell nuclei;

   (d) Nucleolar: large coarse speckled staining within the nucleus, less than six in number per cell;

   (e) Cytoplasmatic: fine fluorescent fibres running the length of the cell;

   (f) Centromere: several discrete speckles (40-60) distributed throughout the inter-phase nuclei and characteristically found in the condensed nuclear chromatin.

   From the above description one can note the fact that both "fine speckled" and "coarse speckled" are having granular nuclear staining of inter-phase cell nuclei. Similarly "homogeneous" is also having staining of inter-phase nucleus though the staining pattern is different. "Centromere" and "homogeneous" both uses staining of nuclear chromatin. Hence there is some inter-class similarity between classes. So we use family specific dictionaries where each family is a cluster of few classes. The "Family Specific" dictionary contains those features which are common between some classes but not common to all classes. We assume there are $\mathbb{F}$ number of families or class-clusters. f^{th} family specific dictionary is given as
$$D_{\mathbb{C}+f} \in \mathbb{R}^{d \times K_{\mathbb{C}+f}} \quad f = 1, 2, \ \ldots \ \mathbb{F}$$

where $K_{\mathbb{C}+f}$ is the number of atoms allowed to the f^{th} Family. So the family specific dictionary is given as

$$D_{family\ \ specific} = \begin{bmatrix} D_{\mathbb{C}+1} & D_{\mathbb{C}+2} & \dots & D_{\mathbb{C}+f} & \dots D_{\mathbb{C}+\mathbb{F}} \end{bmatrix}$$

3. Commonality Dictionary : The images of HEp-2 cells are all captured in fluorescent base. All the images are being cell images, they have lot of similarity among them. So we use a commonality dictionary D_0, $D_0 \in \mathbb{R}^{d \times K_0}$, which stores the common features between all the class specific signals. where K_0 is the number of atoms in commonality dictionary.

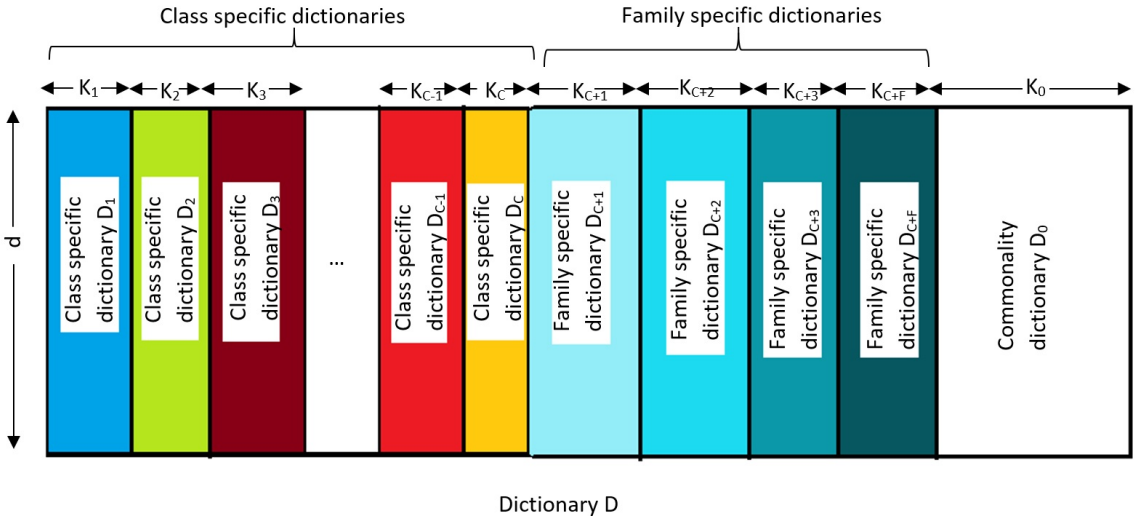

Figure 3.1: Visualizing the dictionary structure

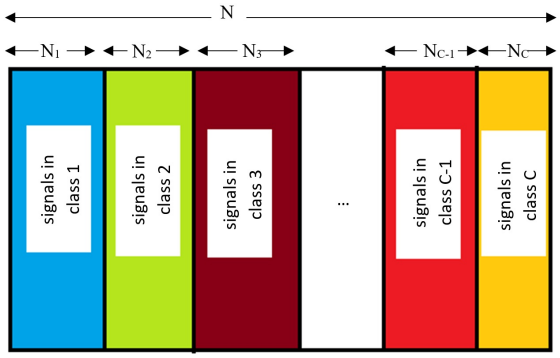

Figure 3.2: Corresponding Y Matrix

$$D = \begin{bmatrix} D_0 & D_1 & D_2 & D_3 & \dots & D_{\mathbb{C}+i} & \dots & D_{\mathbb{C}} & D_{\mathbb{C}+1} & D_{\mathbb{C}+2} & \dots & D_{\mathbb{C}+f} & \dots D_{\mathbb{C}+\mathbb{F}} \end{bmatrix}$$

where $\mathbb{C}$ is the number of classes and $\mathbb{F}$ is the number of family.

The total number of atoms is given as

$$K = K_0 + \sum_{i=1}^{\mathbb{C}} K_i + \sum_{j=1}^{\mathbb{F}} K_{\mathbb{C}+j} \tag{3.1}$$

In our case the objective function is

$$\begin{aligned} \sum_{i=1}^{\mathbb{C}} \Bigg(& \|Y_i - DX_i\|_F^2 + \lambda_1 \Big\| Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \Big\|_F^2 \\ & + \lambda_2 \Big\{ \|X_i - M_i\|_F^2 - \|M_i - M_0\| \Big\} + \lambda_3 \sum_{j=0, j\neq i}^{\mathbb{C}+\mathbb{F}} \left\| D_i^T D_j \right\|_F^2 \Bigg) + \lambda_4 \|X\|_1 \end{aligned} \tag{3.2}$$

number of signal in i^{th} class is given by N_i and the total umber of signals is given by

$$N = \sum_{i=1}^{\mathbb{C}} N_i \tag{3.3}$$

i^{th} class specific signal given by $Y_i \in \mathbb{R}^{d \times K_i}$. Similarly X denotes the overall sparse representation of the signal with respect to the complete dictionary,D. $X \in \mathbb{R}^{\mathbb{K} \times \mathbb{N}}$. The symbol X_i denotes the sparse coefficient of of the signals belonging to class i (Y_i) with respect to the complete dictionary(D), $X_i \in \mathbb{R}^{K \times N_i}$. Similarly the sparse representation of the signal of class i , Y_i over the dictionary D_j us given as $X_i^j \in \mathbb{R}^{K_j \times N_i}$

In a detailed description about the objective function, the first term of the objective function $\|Y_i - DX_i\|_F^2$ signifies that for all class specific signal it must be well approximated by the complete dictionary. In other words, all the signals irrespective of which class it comes from, it must lie in the space spanned by the complete dictionary.

The next term,$\Big\| Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \Big\|_F^2$ comprises of the the three main components of dictionary. If we assume c^{th} class depends on the f^{th} family then we assume that the following approximation holds

$$Y_c \approx D_0 X_c^0 + D_c X_c^c + D_{\mathbb{C}+f} X_c^{\mathbb{C}+f}$$

The term $\|X\|_1$ is the conventional term to add sparsity in the training process. The discriminative fidelity term

$$\sum_{j=0, j\neq i}^{\mathbb{C}+\mathbb{F}} \left\| D_i^T D_j \right\|_F^2$$

is added to reduce the similarity between the different sub-dictionaries. This term has been earlier used by FDDL [52] and other algorithms [50].

The term that remains to be discussed is $\left(\|X_i - M_i\|_F^2 - \|M_i - M_0\|\right)$. This term follows from LRSDL [19] implementation. We define the mean operation

$$\Psi : \mathbb{R}^{\mathbb{M}\times\mathbb{N}} \rightarrow \mathbb{R}^{\mathbb{M}\times\mathbb{N}}$$

as $\psi(A) = \tilde{A}$ where Ψ first takes mean of each row for the A Matrix, and thus we obtain a single column vector $\in \mathbb{R}^M$. The mean vector is stacked N times to form the output matrix as same dimension of input matrix. We define $M_i = \Psi(X_i)$ and $M_0 = \Psi(M_i)$

3.2 Optimizing the Objective Function

There are many updates to be done. We list them

1. Update the class specific dictionary for each class
2. Update the family specific dictionary for each family
3. Update the commonality dictionary
4. Update the family assignment to each class
5. Update the sparse coefficient with respect to class specific dictionaries
6. Update the sparse coefficient with respect to family specific dictionaries
7. Update the sparse coefficient with respect to commonality dictionaries

3.2.1 Update class specific dictionary

In this section we derive the equations needed to update each class's dictionary $D_i \quad i = 1, 2, \ \ldots\ \mathbb{C}$. For updating each class specific dictionary we start by keeping all other sub-dictionaries fixed. The updates are done with respect to one atom at a time. To solve the optimum dictionary matrix partial derivative needs to be performed. Since matrix differentiation with respect to matrix requires tensor calculus of higher order, we avoid complete dictionary update at a time and use atom by atom update, instead. Within the class-specific dictionary of the "class of interest" we set all the atoms, other than the atom of interest as constant. Let the i^{th} class dictionary be given as

$$D_i = \begin{bmatrix} d_1^i & d_2^i & d_3^i & \ldots & d_l^i & \ldots & d_{K_i}^i \end{bmatrix}$$

where K_i is the number of atoms in the i^{th} class. However to extract the l^{th} atom of the i^{th} class dictionary we need to define proper linear transformer matrix. We define matrix transformation T where $T_i \in \mathbb{R}^{K_i \times K}$ as

$$T_i = \begin{bmatrix} t_1^i, & ... & t_j^i & ... & t_{K_i}^i \end{bmatrix}$$

where $T_i \in \mathbb{R}^{K_i \times K}$

$$t_j^i = \Big[\underbrace{0, \ \ ,..., \ 0}_{\sum_{p=0}^{i-1} K_m}, \overbrace{\underbrace{0, \ \ ... \ ,0}_{j-1}, \overset{j}{\overset{\uparrow}{1}}, \underbrace{0, \ \ ... \ ,0}_{K_i-j}}^{K_i}, \ \underbrace{0, \ \ ..., \ \ ,0}_{\sum_{m=i+1}^{\mathbb{C}+\mathbb{F}} K_m}\Big]$$

we also define another matrix transformer $\overline{T}$ where $\overline{T_i} \in \mathbb{R}^{K \times K}$ as

$$\overline{T_i} = \begin{bmatrix} T_0 & T_1 & ...T_{i-1} & 0_{K_i \times K} & T_{i+1} & ... & T_{\mathbb{C}+\mathbb{F}} \end{bmatrix}$$

Now using this linear transformer matrix we can rewrite $X_i^i = T_i X_i$ and $D_i = DT_i^T$. Now using this basic simplifications we can rewrite our objective function as follows,

The update equation for class specific dictionaries can be rewritten as following (excluding all class's dictionaries other than the dictionary under consideration)

$$J(D_c) = \underset{D_c}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\|Y_i - DX_i\|_F^2 \right) + \lambda_1 \left\|Y_c - D_c X_c^c - D_{\mathbb{C}+f} X_c^{\mathbb{C}+f} - D_0 X_c^0\right\|_F^2 + \lambda_3 \sum_{j=0, j\neq c}^{\mathbb{C}+\mathbb{F}} \left\|D_c^T D_j\right\|_F^2 \tag{3.4}$$

However we can only update one atom of the c^{th} class specific dictionary. The dictionary D_c can be spilt in it's atoms as

$$D_c = \begin{bmatrix} d_1^c & d_2^c & d_3^c & ... & d_l^c & ... & d_{K_c}^c \end{bmatrix}$$

We consider the generalised update equation for updating the l^{th} atom i.e. d_l^c The objective functions can be rewritten as [by ignoring those components which only depends on atoms other than d_l^c]

$$J(d_l^c) = \underset{d_l^c}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left\| Y_i - \underbrace{D\overline{T_c}^T \overline{T_c} X_i}_{term \ \ 1} - \underbrace{\tilde{g_c} T_c X_i}_{term \ \ 2} - \underbrace{d_l^c h_l^c T_c X_i}_{term \ \ 3} \right\|_F^2 +$$

$$\lambda_1 \left\| Y_c - \underbrace{\tilde{g}_c T_c X_c}_{term\ \ 4} - \underbrace{d_l^c h_l^c T_c X_c}_{term\ \ 5} - D_{\mathbb{C}+f} X_c^{\mathbb{C}+f} - D_0 X_c^0 \right\|_F^2 + \lambda_3 \left\| \left(d_l^c \right)^T D \tilde{T}_c^T \right\|_F^2$$

term 1: Contribution from other dictionary other thane D_c. $D\overline{T_c}^T$ extracts all other dictionary other than the c^{th} class dictionary. The term $\overline{T_c} X_i$ includes contribution of all other dictionaries other than the i^{th} class's dictionary and their corresponding sparse representation is extracted.
term 2: Contribution from all atoms of c^{th} class other than l^{th} atom
term 3: Contribution from d_l^c
term 4: Contribution from all other atoms of D_c other than d_l^c
term 5: Contribution from d_l^c

Here all the signals that atoms of the D_c other than d_l^c is created as

$$\tilde{g}_c = \sum_{m=1, m \neq c}^{K_c} d_m^c h_m^c$$

where h_m^c is an one dimensional row vector of which m^{th} element is set to 1 and all other element is set to 0. Also $\tilde{g}_c \in \mathbb{R}^{d \times K_c}$ and $h_m^c \in \mathbb{R}^{K_c \times 1}$. However to apply all the formulae and shortcuts derived on frobenius Norm we must convert the objective function to some standard form where we can directly apply the formulae derived in previous section

$$P_i = h_l^c T_c X_i \qquad i \in \{1, 2, , ...\mathbb{C}\} \tag{3.5}$$

$$Q_i = Y_i - D\overline{T_c}^T \overline{T_c} X_i - \tilde{g}_c T_c X_i \qquad i \in \{1, 2, , ...\mathbb{C}\} \tag{3.6}$$

$$R_i = D\overline{T_i}^T \qquad i \in \{1, 2, , ...\mathbb{C}\} \tag{3.7}$$

$$S_c = Y_c - D(\overline{T}_{\mathbb{C}+f}^T \tilde{T}_{\mathbb{C}+f} + \overline{T}_0^T \overline{T}_0) X_c - \tilde{g}_c T_c X_c \tag{3.8}$$

using those substitutions we have

$$J(d_l^c) = \underset{d_l^c}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\|Q_i - d_l^c P_i\|_F^2 \right) + \lambda_1 \|S_c - d_l^c P_c\|_F^2 + \lambda_3 \left\| (d_l^i)^T R_c \right\|_F^2$$

Now we apply the equation results from partial derivative of "frobenius Norm" from equation A.4 and equation A.6 in all the terms

$$\frac{\partial \sum_{i=1}^{\mathbb{C}} \left(\|Q_i - d_l^c P_i\|_F^2 \right)}{\partial d_l^c} = -2 \sum_{i=1}^{\mathbb{C}} Q_i P_i^T + 2 d_l^c \sum_{i=1}^{\mathbb{C}} P_i P_i^T$$

$$\frac{\partial \|S_c - d_l^c P_c\|_F^2}{\partial d_l^c} = -2 S_c P_c^T + 2 d_l^c P_c P_c^T$$

$$\frac{\partial \left\| (d_l^i)^T R_c \right\|_F^2}{\partial d_l^c} = R_c R_c^T$$

Putting all the terms together and rearranging a bit we get

$$d_l^c = (A + \lambda_1 P_c P_c^T + \lambda_3 R_c R_c^T)^{-1} \ (B + \lambda_1 S_c P_c^T) \tag{3.9}$$

where $A = \sum_{i=1, i \neq c}^{\mathbb{C}} P_i P_i^T$ and $B = \sum_{i=1, i \neq c}^{\mathbb{C}} Q_i P_i^T$

3.2.2 Update family specific dictionary

Next we derive the update equation family specific dictionaries $D_{\mathbb{C}+f}$ for $f \in [1, 2, \ \ldots \ \mathbb{F}]$. Let us rewrite the main objective function from equation 3.21 in terms of $D_{\mathbb{C}+f}$

$$J(D_{\mathbb{C}+f}) = \underset{D_{\mathbb{C}+f}}{\operatorname{argmin}} \left(\sum_{i=1}^{\mathbb{C}} \|Y_i - D X_i\|_F^2 \right) + \lambda_1 \left(\sum_{i' \in sub-cluster(f)} \left\| Y_{i'} - D_{i'} X_{i'}^{i'} - D_{\mathbb{C}+f} X_{i'}^{\mathbb{C}+f} - D_0 X_{i'}^0 \right\|_F^2 \right.$$

$$\left. + \lambda_3 \sum_{j=1, j \neq \mathbb{C}+f}^{\mathbb{C}+\mathbb{F}} \left\| D_0^T D_j \right\|_F^2 \right.$$

The dictionary $D_{\mathbb{C}+f}$ can be spilt in it's atoms as

$$D_{\mathbb{C}+f} = \left[d_1^{\mathbb{C}+f} \ \ d_2^{\mathbb{C}+f} \ \ d_3^{\mathbb{C}+f} \ \ \ldots \ \ d_l^{\mathbb{C}+f} \ \ \ldots \ \ d_{K_{\mathbb{C}+f}}^{\mathbb{C}+f} \right]$$

We consider the generalised update equation for updating the l^{th} atom i.e. $d_l^{\mathbb{C}+f}$ The objective functions can be rewritten as [by ignoring those components which only depends on atoms other than $d_l^{\mathbb{C}+f}$]

$$J(d_l^{\mathbb{C}+f}) = \underset{d_l^{\mathbb{C}+f}}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left\| Y_i - D \overline{T_{\mathbb{C}+f}}^T \overline{T_{\mathbb{C}+f}} X_i - \tilde{g}_{\mathbb{C}+f} T_{\mathbb{C}+f} X_i - d_l^{\mathbb{C}+f} h_l^{\mathbb{C}+f} T_{\mathbb{C}+f} X_i \right\|_F^2 +$$

$$\lambda_1 \sum_{i' \in sub-cluster(f)}^{\mathbb{C}} \left\| Y_i' - D_{i'} X_{i'}^{i'} - -D_0 X_{i'}^0 - g_{\tilde{\mathbb{C}+f}} T_{\mathbb{C}+f} X_{i'} - d_l^{\mathbb{C}+f} h_l^{\mathbb{C}+f} \overline{T}_{\mathbb{C}+f} X_{i'} \right\|_F^2 +$$

$$\lambda_3 \left\| \left(d_l^{(\mathbb{C}+f)} \right)^T D \overline{T}_{\mathbb{C}+f}^T \right\|_F^2$$

Here the $i' \in sub - cluster(f)$ signifies that the sum is only taken over those indices or those classes which uses the family under consideration. where $\overline{g}_{\mathbb{C}+f} = \sum_{m=1, m \neq l}^{K_{\mathbb{C}+f}} d_m^{\mathbb{C}+f} h_m^{\mathbb{C}+f}$ Again we need some minor substitutions to reshape this objective function to our known form where we can directly apply the derivative of the frobenius Norm, using results from equation A.1 and equation A.2 The substitutions are

$$P_i = h_l^{\mathbb{C}+f} T_{\mathbb{C}+f} X_i \in \mathbb{R}^{1 \times N_i} \tag{3.10}$$

$$Q_i = Y_i - D \overline{T}_{\mathbb{C}+f}^T \overline{T}_{\mathbb{C}+f} X_i - \tilde{g}_i X_i \tag{3.11}$$

$$R_{\mathbb{C}+f} = D \overline{T}_{\mathbb{C}+f}^T \tag{3.12}$$

$$S_{i'} = Y_{i'} - D(\overline{T}_{i'}^T \overline{T}_{i'} + \overline{T}_{\mathbb{C}+\mathbb{F}}^T \overline{T}_{\mathbb{C}+\mathbb{F}}) - \tilde{g}_l{}^{i'} T_{\mathbb{C}+f} X_{i'} \tag{3.13}$$

Using this substitution the objective function can be modified as in much simpler form:

$$J(d_l^{\mathbb{C}+f}) = \underset{d_l^{\mathbb{C}+f}}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\left\| Q_i - d_l^{\mathbb{C}+f} P_i \right\|_F^2 \right) + \lambda_1 \sum_{i' \in sub_c luster(f)}^{\mathbb{C}} \left(\left\| S_{i'} - d_l^{\mathbb{C}+f} P_{i'} \right\|_F^2 \right) + \lambda_3 \left\| (d_l^{\mathbb{C}+f})^T R_{\mathbb{C}+f} \right\|_F^2$$

Now we apply the result for deriving the partial derivative of frobenius Norm from equation A.4 and equation A.6

$$\frac{\partial \sum_{i=1}^{\mathbb{C}} \left(\left\| Q_i - d_l^{\mathbb{C}+f} P_i \right\|_F^2 \right)}{\partial d_l^{\mathbb{C}+f}} = -2 \sum_{i=1}^{\mathbb{C}} Q_i P_i^T + 2 d_l^c \sum_{i=1}^{\mathbb{C}} P_i P_i^T$$

$$\frac{\partial \sum_{i' \in sub_c luster(f)}^{\mathbb{C}} \left(\left\| S_{i'} - d_l^{\mathbb{C}+f} P_{i'} \right\|_F^2 \right)}{\partial d_l^{\mathbb{C}+f}} = -2 \sum_{i' \in sub_c luster(f)}^{\mathbb{C}} S_{i'} P_{i'}^T \; + \; 2 d_l^{\mathbb{C}+f} \sum_{i' \in sub_c luster(f)}^{\mathbb{C}} P_{i'} P_{i'}^T$$

$$\frac{\partial \left\| (d_l^{\mathbb{C}+f})^T R_{\mathbb{C}+f} \right\|_F^2}{\partial d_l^{\mathbb{C}+f}} = R_{\mathbb{C}+f} R_{\mathbb{C}+f}^T$$

Putting all the terms together and rearranging a bit we get

$$d_l^{\mathbb{C}+f} = (A + \lambda_1 C + \lambda_3 R_{\mathbb{C}+f} R_{\mathbb{C}+f}^T)^{-1} \ (\lambda_1 B + E) \tag{3.14}$$

where $A = \sum_{i=1, i\neq c}^{\mathbb{C}} P_i P_i^T$ and $C = \sum_{i' \in sub_cluster(f)}^{\mathbb{C}} P_{i'} P_{i'}^T$ $B = \sum_{i' \in sub_cluster(f)}^{\mathbb{C}} S_{i'} P_{i'}^T$ $E = \sum_{i=1}^{\mathbb{C}} Q_i P_i^T$

3.2.3 Update commonality dictionary

The update equation of the commonality dictionary is comparatively simpler than the previous versions. We have

$$D_0 = \begin{bmatrix} d_1^0 & d_2^0 & ... & d_l^0 & ... & d_l^{K_0} \end{bmatrix}$$

as usual we ignore all other terms other than those contains D_0 in that term and also we do have the modified objective function as

$$J(D_0) = \underset{D_0}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \|Y_i - DX_i\|_F^2 + \lambda_1 \sum_{i=1}^{\mathbb{C}} \left\|Y_c - D_c X_c^c - D_{\mathbb{C}+f} X_c^{\mathbb{C}+f} - D_0 X_c^0\right\|_F^2 + \lambda_3 \sum_{j=1}^{\mathbb{C}+\mathbb{F}} \left\|D_0^T D_j\right\|_F^2$$

we define

$$\tilde{g}_0 = \sum_{m=1, m\neq l}^{K_0} d_m^0 e_m^0$$

Now considering atom by atom we have

$$J(d_l^0) = \underset{d_l^0}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left\|Y_i - D\overline{T}_0^T \overline{T}_0 X_i - \tilde{g}_0 T_0 X_i - d_l^0 h_l^0 T_0 X_i\right\|_F^2 +$$

$$\sum_{i=1}^{\mathbb{C}} \lambda_1 \left\|Y_c - D_c X_c^c - D_{\mathbb{C}+f} X_c^{\mathbb{C}+f} - \tilde{g}_0 T_0 X_i - d_l^0 h_l^0 T_0 X_i\right\|_F^2 +$$

$$\lambda_3 \left\| \left(d_l^0\right)^T D\tilde{T}_0^T \right\|_F^2$$

Now We need to differentiate this with respect to

$$P_i = h_l^0 T_0 X_i \tag{3.15}$$

$$Q_i = Y_i - D\overline{T}_0^T \overline{T}_0 X_i - \tilde{g}_i T_0 X_i \tag{3.16}$$

$$R_0 = D\overline{T}_0^T \tag{3.17}$$

$$S_i = Y_i - D(T_i^T T_i + T_{\mathbb{C}+f}^T T_{\mathbb{C}+f}) X_i - \tilde{g}_i T_0 X_i \tag{3.18}$$

using those substitutions as shown above we have the following simplified equations as

$$J(d_l^0) = \underset{d_l^0}{\operatorname{argmin}} \sum_{i=1}^{\mathbb{C}} \left(\left\| S_i - d_l^0 P_i \right\|_F^2 \right) + \lambda_1 \left\| (d_l^0)^T R_0 \right\|_F^2 + \sum_{i=1}^{\mathbb{C}} (Q_i - d_l^0 P_i)$$

Now using the results derived in equation A.1 and equation A.2 we have the following results

$$\frac{\partial \sum_{i=1}^{\mathbb{C}} \left(\left\| S_i - d_l^0 P_i \right\|_F^2 \right)}{\partial d_l^0} = -2 \sum_{i=1}^{\mathbb{C}} S_i P_i^T + 2 d_l^0 \sum_{i=1}^{\mathbb{C}} P_i P_i^T$$

$$\frac{\partial \sum_{i=1}^{\mathbb{C}} \left(Q_i - d_l^0 P_i \right)}{\partial d_l^0} = -2 \sum_{i=1}^{\mathbb{C}} Q_i P_i^T \;\; + \;\; 2 d_l^0 \sum_{i=1}^{\mathbb{C}} P_i P_i^T$$

$$\frac{\partial \left\| \left(d_l^0 \right)^T R_0 \right\|_F^2}{\partial d_l^0} = R_0 R_0^T$$

Equating the partial derivatives to zero and rearranging a bit we get

$$d_l^0 = \left((1 + \lambda_1) A + \lambda_3 R_0 R_0^T \right)^{-1} (\lambda_1 B + C) \tag{3.19}$$

where $A = \sum_{i=1}^{\mathbb{C}} P_i P_i^T \;\; B = \sum_{i=1}^{\mathbb{C}} Q_i P_i^T \;\; C = \sum_{i=1}^{\mathbb{C}} S_i P_i^T$

3.2.4 Update Family assignment for each class

For each class we need to assign it to a single agglomeration of cluster. There may be many a way to perform that assignment. In this literature we assign each class to, one by one, to all the families and calculate the resulting reconstruction error. Then we sort families according to reconstruction error generated. The family with the least reconstruction error is selected to be the family of the class. For c^{th} class the family is given as f_c and we obtain f_c as

$$f_c = \underset{f}{\operatorname{argmin}} \left\| Y_c - D_c X_c^c - D_{\mathbb{C}+f} X_c^{\mathbb{C}+f} - D_0 X_c^0 \right\|_F^2 \tag{3.20}$$

where $f \in [1, 2, 3, \ \ldots \ \mathbb{F}]$

3.2.5 Update sparse representation with respect to class specific dictionary

Their are four occurrences of sparse coefficient with respect to class specific dictionary in equation 3.21. We write the relevant equation as

$$\begin{aligned} \sum_{i=1}^{\mathbb{C}} \Bigg(\underbrace{\|Y_i - DX_i\|_F^2}_{term\ 1} + \lambda_1 \underbrace{\left\| Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \right\|_F^2}_{term\ 2} \\ + \lambda_2 \underbrace{\Big(\|X_i - M_i\|_F^2 - \|M_i - M_0\| \Big)}_{term\ 3} \Bigg) + \lambda_4 \underbrace{\|X\|_1}_{term\ 4} \end{aligned} \tag{3.21}$$

Now we have for term 1 we apply differentiation rule for frobenius norm from equation A.1 and A.2. We obtain

$$\frac{\partial \Big(\|Y_i - DX_i\|_F^2 \Big)}{\partial X_i} = -2D^T Y_i + D^T D X_i \tag{3.22}$$

similarly for partial derivative of term 2, we can use the expression derived in linear transformer matrix, $X_i^i = T_i X_i$. Thus we have

$$\left\| Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \right\|_F^2 = \left\| Y_i - D_i T_i X_i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \right\|_F^2 = \|V_i - D_i T_i X_i\|_F^2$$

where $V_i = Y_i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0$

$$\frac{\partial \|V_i - D_i T_i X_i\|_F^2}{\partial X_i} = -2T_i^T D_i^T V_i + 2T_i^T T_i X_i \tag{3.23}$$

For the third term, We have to express the mean operation as a linear transformation to perform the partial derivative.Let E_p^q be a matrix $\in \mathbb{R}^{p\times q}$. Thus we obtain

$$M_i = \tfrac{1}{N_i} E_K^{N_i} X_i \text{ since, } X_i \in \mathbb{R}^{K\times N_i}$$

$$X_i - M_i = X_i - \frac{1}{N_i} E_K^{N_i} X_i = X_i (I - \frac{1}{N_i} E_K^{N_i})$$

$$\begin{aligned}\frac{\partial \|X_i - M_i\|_F^2}{\partial X_i} &= \frac{\partial}{\partial X_i}\left[X_i\left((I - \frac{1}{N_i}E_K^{N_i}\right)X_i^T\left(I - \frac{1}{N_i}E_K^{N_i}\right)\right] \\ &= 2X_i\left(I - \frac{1}{N_i}E_K^{N_i}\right)\left(I - \frac{1}{N_i}E_K^{N_i}\right)^T \\ &= 2X_i\left(I - \frac{1}{N_i}E_K^{N_i}\right) \\ &= 2X_i - 2M_i\end{aligned}$$

So we have

$$\frac{\partial \|X_i - M_i\|_F^2}{\partial X_i} = 2X_i - 2M_i \tag{3.24}$$

We also have

$$M_i - M = \frac{1}{N_i} X_i E_K^{N_i} - \frac{1}{N_i} M_i E_K^{N_i}$$

thus we get

$$\frac{\partial \|M_i - M\|_F^2}{\partial X_i} = \frac{1}{N_i}\frac{\partial X_i}{\partial X_i} E_K^{N_i} - \frac{1}{N_i}\frac{\partial M_i}{\partial X_i} E_K^{N_i}$$

$$\begin{aligned}&= \frac{1}{N_i} E_K^{N_i} - \frac{1}{N_i}\big(2X_i - 2M_i\big) E_K^{N_i} \\ &= \frac{1}{N_i} E_K^{N_i} - \big(2M_i - 2M\big)\end{aligned}$$

So we have,

$$\frac{\partial \|M_i - M\|_F^2}{\partial X_i} = \frac{1}{N_i} E_K^{N_i} - \big(2M_i - 2M\big) \tag{3.25}$$

For term 4 we have $X_i^i = T_i X_i$ and $X_i = \overline{T}_i^T \overline{T}_i X$. So we express X_i as a linear transformed product of X we have $X_i = T_i X$. This implies, $X = \Psi_i X_i^i$, where $\Psi_i = \left(\overline{T}_i^T \overline{T}_i\right)^{-1}$. So we can rewrite the fourth term as

$$\|X\|_1 = \left\| \Psi_i \left(X_i \right) \right\|_1$$

Now to differentiate the l_1 norm their issue as l_1 norm is not differentiable at zero. So we use the Iterative Shrinckage Thresholding Algorithm(ISTA) [4]. Which uses for each dimension j in the argument vector the result of partial derivative is $+1$ if value is > 0 else negative. An if the gradient changes sign then it indicates the terminal case of no differentiability at 0 and we clamp the gradient to zero.

$$\frac{\partial \|X\|_1}{\partial X_i} = \frac{\partial \left(\|\Psi_i X_i\|_1 \right)}{\partial X_i} = \|\Psi_i\|_1 * \frac{\partial \left(\|X_i\|_1 \right)}{\partial X_i} = \|\Psi_i\|_1 * \begin{cases} +1, & \text{if } X_i^j > 0 \\ -1, & \text{if } X_i^j < 0 \\ 0, & \text{if } X_i^j = 0 \end{cases} \tag{3.26}$$

where j varies from 0 to number of dimensions in argument X_i. For simplicity of symbols we denote the opration defined in equation 3.26, as $\frac{\partial \|X\|_1}{\partial X_i} = ISTA(X_i)$

Thus combining partial derivative from equations 3.22, 3.23, 3.24 and 3.26 we obtain the final update equation for X_i as

$$-2D^T Y_i + 2D^T D X_i + \lambda_1 \left(-2T_i^T D_i^T V_i + 2T_i^T T_i X_i \right) + \lambda_2 \left[(2X_i - 2M_i) + \frac{1}{N_i} E_K^{N_i} - \left(2M_i - 2M \right) \right]$$
$$+ \lambda_3 \|\Psi_i\|_1 * ISTA(X_i) = 0$$

$$\implies -2D^T Y_i + 2D^T D X_i + \lambda_1 \left(-2T_i^T D_i^T V_i + 2T_i^T T_i X_i \right) + \lambda_2 \left[(2X_i - 2M) + \frac{1}{N_i} E_K^{N_i} \right]$$
$$+ \lambda_3 \|\Psi_i\|_1 * ISTA(X_i) = 0$$

rearranging a bit we obtain

$$X_i = \left(2D^T D + \lambda_1 2T_i^T T_i + 2\lambda_2 I \right)^{(-1)} \left(2D^T Y_i + 2\lambda_1 T_i^T D_i^T V_i \right.$$
$$\left. \lambda_2 \left(2M + \frac{1}{N_i} E_K^{N_i} \right) + \lambda_3 \|\Psi_i\|_1 * ISTA(X_i) \right) \tag{3.27}$$

3.2.6 Update Sparse representation for each family specific dictionary

Similar to the discussion above we have, The relevant objective function is

$$J(X_i^{\mathbb{C}+f}) = \left\| Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \right\|_F^2$$

We do some minor substitution to transform this to a more convenient form where we can directly apply the formulae of differentiating the frobenius Norm Let $V_{\mathbb{C}+f} = Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0$ Now using that substitution we do have substitution we get

$$J(X_c^i) = \left\| V_{\mathbb{C}+f} - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} \right\|_F^2$$

Now using the results about derivative of frobenius Norm from equation A.1 and equation A.2 we get

$$\frac{\partial \left\| V_{\mathbb{C}+f} - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} \right\|_F^2}{\partial X_c^c} = 0$$

$$X_i^{\mathbb{C}+f} = (D_{\mathbb{C}+f}^T D_{\mathbb{C}+f})^{-1} (D_{\mathbb{C}+f}^T V_{\mathbb{C}+f}) \tag{3.28}$$

For each class c we calculate this $X_i^{\mathbb{C}+f}$ for $i \in [1, 2 \ ... \ \mathbb{C}]$ and stack all such $X_i^{\mathbb{C}+f}$ to get final $X^{\mathbb{C}+f}$

3.2.7 Update Sparse representation for commonality dictionary

Similar to the discussion above we have, The relevant objective function is

$$J(X_i^0) = \left\| Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0 \right\|_F^2$$

We do some minor substitution to transform this to a more convenient form where we can directly apply the formulae of differentiating the frobenius Norm Let $V_0 = Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f}$ Now using that substitution we get

$$J(X_i^0) = \left\| V_0 - D_0 X_i^0 \right\|_F^2$$

Now using the results about derivative of frobenius Norm from equation A.1 and equation A.2 we get

$$\frac{\partial \| V_0 - D_0 X_i^0 \|_F^2}{\partial X_i^0} = 0$$

$$X_i^0 = (D_0^T D_0)^{-1} (D_0^T V_0) \tag{3.29}$$

For each class c we calculate this X_i^0 for $i \in [1, 2 \ ... \ \mathbb{C}]$ and stack all such X_i^0 to get final X^0

3.3 Pre-processing

We tried with different approaches in pre-processing during developement of our algorithm. "Dictionary Learning" algorithm considers signals in one dimensional vector form. So we have to linearize the two dimensional image. Simplest way to perform that was to stack columns in the images one after another to form a single column vector of signal. However this was not a successful idea. The reason being, by stacking one column after another column together we were simply losing all the information about the neighbourhood property. Thus important features in the image were lost.

However to get better result we started with patch based image processing. We resorted to fully overlapping patches. The patches had 50 % overlap in horizontal direction as well as 50% overlap in vertical direction. The fully overlapping patch based approach was good at capturing information. However, use of fully overlapping patches, resulted increased number of dimension of final signal vector. This lead to a proportionate increase in in computational cost of the algorithm. As already discussed, dictionary learning systems are designed to be over-complete. In other words, the number of dimension of the signal should be less than the number of atoms in the dictionary. However in worst case, if we consider the number of dimension to be equal to number of atoms, still the overall algorithm is proportional to dimension, raised to the power of 5. Hence increasing the number of dimensions has negative effect on computational cost. Using fully overlapping patches provided higher classification accuracy, but computational cost still was not acceptable in comparison to other state-of-the-art algorithms.

Next we tried not to use the raw images as input to the algorithm. Instead we used texture extractor filters, MR-8. MR-8 filter consists of 38 different filters. It incorporates a Gaussian and a Laplacian of Gaussian(LoG) both at scale $\sigma = 10$ pixels. They are an-isotropic filters. Then we have a bar filter and an edge filter both at 3 scales and 6 directions.The scales are $(\sigma_x,\ \sigma_y)$=$\{(1,3),(2,6),(4,12)\}$. The six degrees of orientations are 0°, 30°, 60°, 90°, 120°, 150° For the two an-isotropic filter their response is taken. For bar and edge filter both, for each of the scale, we take maximum along that scale of all six direction . We have two responses for three scales for bar and edge filter. Altogether we have 8 total responses in the final output.

However, this effort did not give good results. Next we tried to use the local feature extractors like Sift Invariant Feature Transform (SIFT) [24] and the SURF [3]. By using these feature extractors, we tried to address two major drawbacks of the previous approaches. The first drawback was, when we are using patch based processing we were giving equal importance to all the patches. All the patches did not contain necessary information. Secondly the dimension of the image, as we have discussed, is causing negative effect in computational time. In our implementation, SURF was faster than SIFT, so we use SURF. Since local feature extractors like SURF

or SIFT does not gives equal importance to all the patches, only relevant patch area used. So, this removes the first drawback. We intentionally have taken the top 10 features in each image. To select the top ten features we have adaptively changed the Hessian threshold in the SURF detector. For each feature, it's corresponding feature descriptor in SURF can be 64 dimensional or 128 dimensional. We have taken the 64 dimensional feature descriptor.

IN MIVIA HEp-2 cell database we used the masks provided. The masks were prepared by human expert to segment cell image. We used those masks in our algorithm. And again the HEp-2 images, though had 3 channels-RGB, but very little information was contained in the Red and Blue channel. Thus all the image processing tasks in our algorithm were done on the single channel image(Green channel).

3.4 Initialization

Proper initialization schemes are instrumental to obtaining high accuracy of the algorithms. Different literature has suggested different initialization schemes for their dictionary initialization. Complete random dictionary initialization is computationally inexpensive way to initialize the dictionary. But in our case we have different sub-dictionaries for different purposes. Using the same random initialization for all the sub-dictionaries of the dictionary did not prove to provide good results and provide quick saturation of accuracy and loss.

The authors like [51], [50], [7] used an initialization where for c^{th} class they have used the signals with c^{th} class label and randomly used some of those signals as initial value of dictionary atoms. However in recent years, LRSDL reported by Monga *et.al.* [47] uses Online Dictionary Learning [26] algorithm for initializing all it's class specific as well as well as commonality dictionary. A few iterations of ODL was computationally expensive but it provided really good accuarcy. We being inspired by these ideas, tried with different algorithms to initialize our proposed dictionary, simultaneously keeping the computational complexity as low as possible. We have used K-SVD [1] for class specific as well as other sub-dictionary initialization. For each class specific dictionary we run a few iterations of K-SVD only on those class specific signals. The output dictionary of the K-SVD is used as initial dictionary for our algorithm. For commonality dictionary initialization we run K-SVD on the whole signal set. For family specific dictionary initialization, we used a number of approaches like using spectral clustering. None of them provided good result. We randomly assigned few classes to each family and executed K-SVD on those classes of signals to initialize the family specific dictionary.

3.5 Classification Stage

We have tried different approaches in classification. We first tried with most popular classification scheme based on the reconstruction error. Once we have learned the class specific dictionary, we one by one assume the test signal to belong to each class and evaluate the reconstruction error for that class. During train we have preserved the information that each class belongs to some specific family.

However it could not provide good accuracy. The major problem with biomedical data-set is we don't have equal number of patients from all the data class. Some classes of ailments are quite common and some classes are very rare. So the reconstruction error based classification was obviously biased towards the strongest representative class. Classes with many examples will have a well trained dictionaries and may have very low variation in reconstruction error. However the weak representation class's signals do have few examples to train and have very high reconstruction error for even signals belonging to their class also. So we didn't use the reconstruction error based classification concept.

Signals belonging to a particular class will use mainly the dictionary atoms from the sub-dictionary for that particular class. This is the central idea behind our proposed classification scheme. So if we use a separating hyper-plane on the spanning space of the dictionary atoms, we can classify each test signal better. However the commonality and family specific atoms are excluded from classifier training. So once the training is done we extract the relevant columns of the sparse coefficient matrix X, only with respect to sub-dictionaries corresponding to the dictionaries of the class specific dictionaries. Then we train a SVM on the the reduced X_R matrix, we use that trained SVM to classify any new test signal.

3.6 Algorithm

The training algorithms flowchart can be written as

Algorithm 3 DDLICD training

Input: Training signal Y
Output: Dictionary D Sparse representation X
Initialization: Initialize the dictionary by executing the K-SVD algorithm

```
/* comments on code                                                    */
while iteration<= Iteration_MAX do
    for class<= ℂ do
        update D_c using equation 3.9
         update X_c using equation 3.27
         update class vs family assignment using equation 3.20
    end
    for family<= 𝔽 do
        update D_{ℂ+family} using equation 3.14
         update X_{ℂ+family} using equation 3.28

    end
    update commonality dictionary D_0 using equation 3.19
     update sparse representation with respect to commonality dictionary X_0 using
     equation 3.29
     reshape sparse coefficient matrix X to obtain X_reduced
     train the SVM for classification
end
```

3.7 Complexity Analysis

In this section we analyse our algorithm's computation cost. We have assumed that, for a n row matrix we calculate it's inverse in $\mathcal{O}(n^3)$ complexity. Also though their are strassens algorithm which performs matrix multiplication in optimized complexity, we in our case assume that for $A_{m\times n} \times B_{n\times q}$ form of matrix multiplication we have the computational cost as $\mathcal{O}(mnp)$. Since dictionary learning algorithms explicitly make the dictionary over-complete. So we can assume that $K \approx d$, where d is the dimension of the signala nd K is the number of atoms in the dictionary. Also we safely assume number of atoms in the sub-dictionaries in our algorithm is constant fractions of total number of atoms in the complete dictionary. Thus for i^{th} sub-dictionary, it's number of atoms, $k_i = \frac{1}{c_{k_i}}K$ where c_{k_i} is a non zero positive integer. Similarly we assume number of signals with i^{th} class label, $N_i = \frac{1}{c_{n_i}}N$ where c_{n_i} is again a non zero positive integer and N is the total number of signals.

We first evaluate the computational complexity of the class specific dictionary update equation.We evaluate the complexity of equations 3.5, 3.6, 3.7, 3.8

$$\underbrace{P_i}_{1\times N_i} = \underbrace{h_l^i}_{1\times K_i}\underbrace{T_i}_{K_i\times K}\underbrace{X_c}_{K\times N_c} \in \mathbb{R}^{1\times N_i}$$

$$Q_i = \underbrace{Y_i}_{d\times N_i} - \underbrace{D}_{d\times K}\underbrace{\overline{T_i}^T}_{K\times K}\underbrace{\overline{T_i}}_{K\times K}\underbrace{X_i}_{K\times N_i} - \underbrace{\tilde{g}_i}_{d\times K_i}\underbrace{T_i}_{K_i\times K}\underbrace{X_i}_{K\times N_i}$$

$$R_i = \underbrace{D}_{d\times K}\underbrace{\overline{T_i}^T}_{K\times K}$$

$$S_c = \underbrace{Y_c}_{d\times N_c} - \underbrace{D}_{d\times K}(\underbrace{\overline{T}_{\mathbb{C}+f}^T}_{K\times K}\underbrace{\overline{T}_{\mathbb{C}+f}}_{K\times K} + \underbrace{\overline{T}_0^T}_{K\times K}\underbrace{\overline{T}_0}_{K\times K})\underbrace{X_c}_{K\times N_c} - \underbrace{\tilde{g}_c}_{d\times K_c}\underbrace{T_c}_{K_c\times K}\underbrace{X_c}_{K\times N_c}$$

For obtaining P_i we multiply h_c^l with dimension $[1 \times K_i]$ with T_i with dimension $[K_i \times K]$ and X_i with dimension $[K \times N_i]$. This leads to overall computation complexity of $\mathcal{O}(K_iKN_i + K_iN_i)$. However using the result $K_i = \frac{1}{c_{k_i}}K$ and $K_iKN_i \gg K_iN_i$, we simplify the result as $\mathcal{O}(K^2N)$. Similarly we can get computational cost for calculating Q_i as $\mathcal{O}(K^2N_i + K^2N_i + dKN_i)$ However using $K \approx d$ we have $\mathcal{O}(K^2N)$. Similarly R_i evaluation we have $\mathcal{O}(dK^2) \approx \mathcal{O}(K^3)$. Finally, we have for S_c calculation we have $\mathcal{O}(K^3 + K^2N)$. So overall we have complete complexity is $\mathcal{O}(K^3 + K^2N)$. Next we evaluate the complexity of the following two operation, evaluating A and B and finally evaluating d_l^c from equation 3.9

$$d_l^c = (A + \lambda_1 P_c P_c^T + \lambda_3 R_c R_c^T)^{-1}\ (B + \lambda_1 S_c P_c^T)$$

where $A = \sum_{i=1,i\neq c}^{\mathbb{C}} P_i P_i^T$ and $B = \sum_{i=1,i\neq c}^{\mathbb{C}} Q_i P_i^T$. Calculation of matrix A involves the transpose of a matrix P_i and summation over number of classes $\mathbb{C}$. Transpose of matrix P is $\mathcal{O}(N^2)$ (Since P matrix's dimension is is $1 \times N_i$) and multiplication of P and P^T also have same complexity of $\mathcal{O}(N^2)$. So the computational complexity for calculating A is evaluated is $\mathcal{O}(\mathbb{C}N^2 + \mathbb{C}K^2N)$. Similarly the complete cost for evaluation of B is $\mathcal{O}(\mathbb{C}K^3 + \mathbb{C}K^2N)$. Also we have, multiplication of $P_cP_c^T$ calculation has $\mathcal{O}(N^2)$ cost and cost for calculating $R_cR_c^T$ is also same as $\mathcal{O}(K^3)$. The complexity of evaluation of the first inverse matrix of the expression of d_l^c is evaluated as $\mathcal{O}(K^3)$. So the overall complexity is for each atom of each class specific dictionary is dictionary update is $\mathcal{O}(\mathbb{C}K^3 + \mathbb{C}K^2N + \mathbb{C}N^2)$. However this is update equation of a single atom of a single class specific dictionary. For all the atoms of all the class specific atoms we have to multiply the whole computation $\mathbb{C}$ times K. This updates the cost expression as $\mathcal{O}(\mathbb{C}^2K^4 + \mathbb{C}^2K^3N + \mathbb{C}^2KN^2)$. However, for biomedical image processing like HEp 2 cell classification or diabetic retinopathy as in our case ,the number of class is really small. So we can neglect the dependency on number of class or number of family. Hence this step complexity simplifies to

$$\mathcal{O}(K^4 + K^3N + KN^2)$$

Same form of update equations are used for the family specific dictionary update and commonality dictionary update. So the overall computational complexity remains invariant due to their contributions. Family assignment for each class is computationally cheap operation. So we ignore its computational cost.

Analysing computational cost for updation of sparse representation for different dictionaries is relatively simpler than previous analysis. Equation 3.27 has the final update equation for sparse representation update for class specific dictionaries. To evaluate $D^T D$ cost is $\mathcal{O}(K^3)$. Next for $T_i^T T_i$ the cost is $\mathcal{O}(k^3)$ too. Similarly we have the cost for $2D^T Y_i$ the cost is $\mathcal{O}(K^2 N)$. M_i and M are obtained in in same dimension of reading the X matrix. So, their cost is $\mathcal{O}(KN)$. So overall we have complexity of X_c update is as $\mathcal{O}(K^2 N + KN + K^3)$. This when added to computational cost the overall cost remains same. So the overall cost is

$$\mathcal{O}(K^4 + K^3 N + KN^2)$$

Chapter 4

Results and Discussion

We have tested our algorithm on three widely accepted benchmark data-set of HEp-2 cell. The data set published in ICPR 2012 [13], data-set of ICIP 2013 [*] an SNP dataset. All the experiments were conducted partly on a Intel Core(TM) i5 (3.40 GHz) PC with 8 GB of RAM and Windows 10 operating system and some parts of the results are executed on Intel Xenon Sever E7-8890 (2.5 GHz) with 32 core CPU. On each data-set we compare our algorithm with two groups of algorithms. The first group of algorithms contains relevant dictionary learning algorithms. The second group comprises of respective winners and best performer in those competitions. In dictionary learning algorithm comparison we compare these four major algorithms published in recent years, DL-COPAR [19], FDDL [53], D-KSVD [20], LRSDL [47]. The details of these algorithms have been discussed in the literature review section of this paper.

4.1 Comparison with respect to ICPR 2012 data-set

HEp-2 cell data-set from ICPR 2012 was published in two formats. The first data-set contains 1457 individual cell images with corresponding class labels. The cell level data-set is split into two sets by the competition organizers, a train set with 723 images and a test set containing 734 images. The 6 classes of interest are, "centromere", "coarse-speckeled", "cytoplasmatic", "fine speckled", "homogeneous" and "nucleolar". Each individual image is annotated and segmented manually by experts. Images from this database, are acquired by means of a fluorescence microscope (40-fold magnification) coupled with a 50-W mercury vapor lamp and a digital camera utilizing a CCD with square pixel of 6.45 m. Resolution of the images is, 1,388×1,038 pixels and it has 24 bits of color depth. Also, for each image, a fluorescent intensity

[*]https://mivia-web.diem.unisa.it/contest-icip-2013/

tag, i.e., "positive intensity" or "intermediate intensity", is assigned, however those intensity tags are not of practical use to our algorithm.

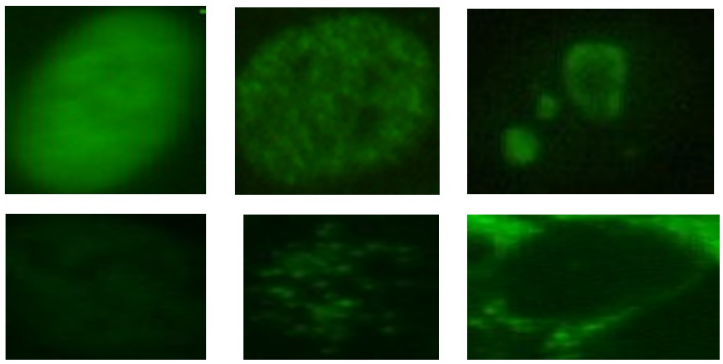

Figure 4.1: ICPR 2012 data-set images of different classes viz-homogeneous, fine speckled,coarse speckled, cytoplasmatic,centromer,neucleolar in clockwise direction from top left corner, respectively

The distribution of test and train images of in different classes is shown in table 4.1

Cell Level classification data		
	Train	Test
centromere	208	149
coarse-speckeleds	109	101
cytoplasmatic	60	51
fine speckled	94	114
homogeneous	150	180
nucleolar	102	139
total	723	734

Table 4.1: Cell level data for ICPR 2012

The confusion matrix of our algorithm is as shown

	Prediction						
		homogeneous	coarse-speckeled	fine speckled	nucleolar	centromere	cytoplasmatic
Ground Truth	homogeneous	80.6	1.7	12.5	1.8	3.3	0.1
	coarse-speckeled	4.1	65.2	6.8	1.0	21.9	1.0
	fine-speckled	22.6	17.9	36.4	0.5	20.4	1.8
	nucleolar	6.5	4.3	0.1	69.5	16.5	3.1
	centromere	0.5	1.3	0.2	9.4	88.1	0.5
	cytoplasmatic	3.3	3.0	2.8	3.6	6.2	80.1

Table 4.2: Cell level confusion matrix for ICPR 2012

The average accuracy is evaluated by calculating average of all the class's accuracy. The average accuracy obtained by our algorithm on cell level classification data on ICPR 2012 data set is 70.02%. Table 4.6 evidently shows that for fine-speckeled class identification algorithm gets frequently confused with other classes like homogeneous and centromere. However, even though cytoplasmatic is the class with least number of cells in training set still our algorithm is able to classify it with high accuracy. Hence this justifies the success of our SVM based spatial decomposition classification scheme which was specially designed to avoid unncessary bias due to class imbalance. Also coarse speckled gets high miss-classification error from the class centromere. This follows from the fact that their is high amount of between class similarity in HEp2 cell classes.

We compare our algorithms performance with other state-of-the-art dictionary learning algorithms. These dictionary learning algorithms have been studied in details in section 2.

Cell Level Accuracy	
Algorithm	accuracy
DL-COPAR [19]	57.34
FDDL [53]	61.1
D-KSVD [20]	64.98
LRSDL [47]	67.7
Our Algorithm	**70.02**

Table 4.3: Comparison with other dictionary learning algorithms

As already discussed in section 3, the major difficulty in effective classification of HEp-2 cell is its inter-class similarity and intra-class variations. The existing dictionary learning algorithms try to make class-specific dictionaries highly discriminative. Hence they easily lose inter-class features. However LRSDL [47] has dedicated commonality dictionary and hence achieved closely second rank in search of best accuracy. Table 4.3 justifies the effectiveness of our proposed family specific dictionary structure or the inter-class clustering concept. The accuracy vs epoch plot for different dictionary learning algorithms are shown in figure 4.2

Next, we compare with other state-of-the-art algorithms including the winners of ICPR 2012.

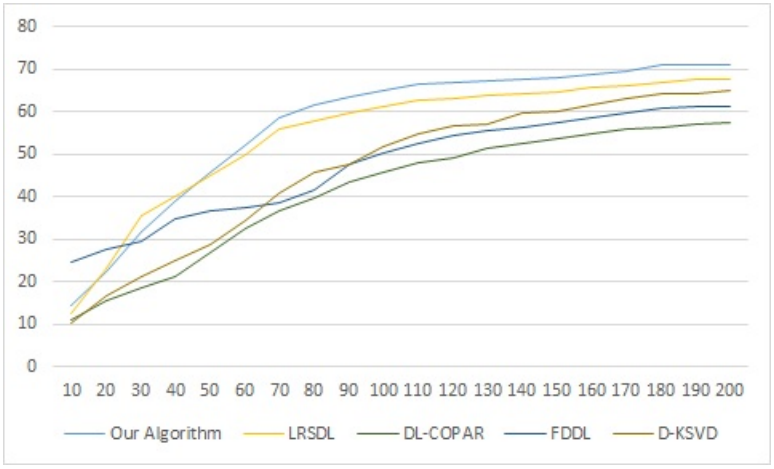

Figure 4.2: Convergence plot for dictionary learning algorithm

Cell Level Accuracy	
Methods	accuracy (%)
Li et al. [23]	64.2
Nokasa et al.[29]	68.7
GoC-LBPs [45]	69.2
SIFT (VLAD) [18]	70.2
Shape Index Histograms [21]	**74.5**
PRICoLBP [41]	**79.6**
RootSIFT [40]	**75.4**
Our Algorithm	**70.02**

We observe from Table 4.1 the folllowing. The PRICoLGBP[41], RootSIFT[40], Shape Index Histograms [21] and our algorithm, report higher accuracy on this data-set. These algorithms outperform the winner of the ICPR 2012 contest, Nokasa et al.[29]. However PRICoLBP and RootSIFT, as discussed in section 3.2, is computationally complex algorithm, Shape index Histogram is another codebook based algorithm and is marginally poor performance than our algorithm. In this context, is worth mentioning that, the task is even challenging for the human expert. Human accuracy is of 73.3% achieved when no other information is given, like neighbourhood cell labels. From this perspective, an accurate HEp-2 cell classification system is very valuable.

The second set of images from MIVIA ICPR 2012 contains the image level data where the complete slide as obtained for a patient, is supplied. The image level data-set is given along with relevant segmentation masks for each slide. Some sample images are shown in Figure **??** and 4.5.

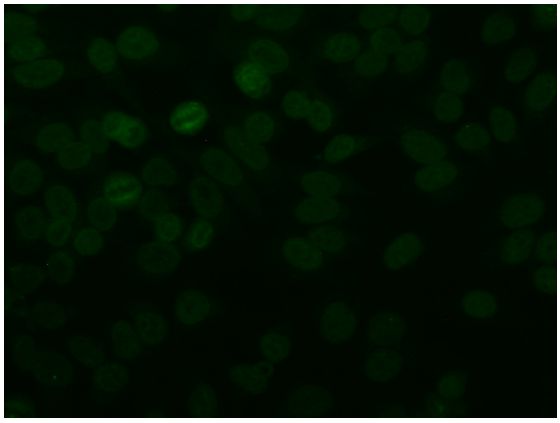

Figure 4.3: Image

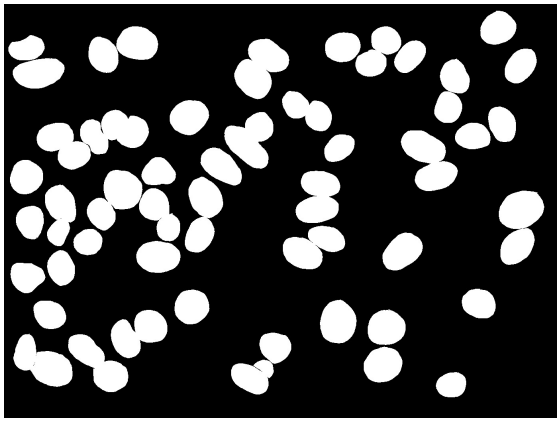

Figure 4.4: Mask

It should be noted that. the class labels corresponding to each image slide, is assigned to strongest representative class. The organisers proposed to test on the slides using a leave-one-out protocol. In other words train on 27 slides and the classifier is tested on the 28^{th} slide. The slide IDs and corresponding class labels are given as

ID	class-label	number of cells	ID	class-label	number of cells
#1	homogeneous	61	#15	fine-speckled	63
#2	fine-speckled	48	#16	Centromere	38
#3	Centromere	89	#17	coarse-speckeled	19
#4	nucleolar	66	#18	homogeneous	42
#5	homogeneous	47	#19	Centromere	65
#6	coarse-speckeled	68	#20	nucleolar	46
#7	Centromere	56	#21	homogeneous	61
#8	nucleolar	56	#22	homogeneous	119
#9	fine-speckled	46	#23	fine-speckled	51
#10	coarse-speckeled	33	#24	nucleolar	73
#11	coarse-speckeled	41	#25	cytoplasmatic	24
#12	coarse-speckeled	49	#26	cytoplasmatic	36
#13	Centromere	46	#27	cytoplasmatic	38
#14	Centromere	63	#28	cytoplasmatic	13
Total	1457				

Table 4.4: ICPR 2012 data-set

The confusion matrix of our proposed algorithm on image level data set of ICPR 2012 is given in table 4.5

		Prediction					
		Ccentromere	coarse-speckeled	fine speckled	homogeneous	nucleolar	cytoplasmatic
Ground Truth	centromere	81.4	0.3	0.5	2.7	14.6	0.5
	coarse-speckeled	0.2	81.2	14.8	0.6	1.8	0.4
	fine speckled	2.6	0.9	65.3	0.0	31.2	0.0
	homogeneous	3.2	2.7	11.9	80.6	0.0	1.7
	nucleolar	1.1	1.8	0.0	6.4	90.7	0.0
	cytoplasmatic	0.0	3.1	0.0	0.8	0.0	96.1

Table 4.5: Image level confusion matrix for ICPR 2012

On image level data-set we could achieve an accuracy of 82.55%. Evidently it achieved higher accuracy achieved than that of cell level image for the same data-set. The results on image level utilizes that in a slide the reported class has always the maximum representation. Hence even if a few cell is mis-classified still the overall class label can be accurately achieved unless the miss-classified cells do not become majority in a slide. All the results reported in the paper are evaluated for 10 independent execution and then averaged and reported. Comparison report to other dictionary learning algorithms, is tabulated as follows

Image Level Accuracy	
Algorithm	accuracy
DL-COPAR [19]	72.1
FDDL [53]	75.8
D-KSVD [20]	78.9
LRSDL [47]	80.7
Our Algorithm	82.55

Similarly we compare with other state-of-the art algorithms.

Cell Level Accuracy	
Methods	accuracy (%)
Li et al. [23]	78.4
Nokasa et al.[29]	81.7
GoC-LBPs [45]	80..2
SIFT (VLAD) [18]	78.1
Shape Index Histograms [21]	80.5
PRICoLBP [41]	**90.2**
RootSIFT [40]	**88.6**
Our Algorithm	**82.55**

4.2 Comparison on ICIP 2013 data-set

The ICIP 2013 data-set used 419 patients positive serum with screening dilution 1:80. The specimens were automatically captured using a cooled microscope with high dynamic range(monochromatic). In total, there are 68,429 extracted cell images. The

whole cell image sets were divided into two sets, according to the experimental protocol. First 13,596 cell images constitute training samples and second set of 54,833 cell images are testing samples. The organizer did not publish the test set for researchers. Hence only train set images are used for both training and testing. A ground-truth mask image is also provided along with each cell image. Images of cells were categorized into six classes: "homogeneous", "speckled", "nucleolar", "centromere", "nuclear membrane", and "golgi". The data-set includes two patterns less frequent occurring in the practical clinic, which are "nuclear membrane" pattern and "golgi" pattern. Thus, it offers a more realistic evaluation on the automatic classification algorithms than the earlier data set. Since only the train image set is available, We partitioned the training image set of 13,596 images into a training set consisting of "6,842" cell images from 42 slides and a test set consisting of "6,754" cell images from 41 slides.

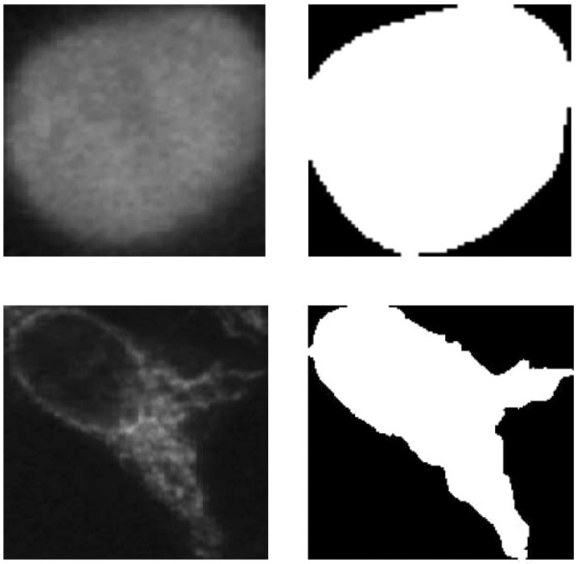

Figure 4.5: Images from ICIP 2013

In ICIP 2013 data-set the class lables are diffeent from ICPR 2012 data-set. Information for the training data of ICIP 2013 contest is shown in Table 4.2.

ICIP 2013 data-set	
class	No. of images
Homogeneous	2494
Speckled	2831
Neucleolar	2598
Centomere	2741
Nuclear Membrane	2208
Golgi	724
Total	13596

The test train split in our case is as follows(42 slides in train and 41 slides in test)

ICIP 2013 data-set		
class	Test image	train image
Homogeneous	1347	1147
Speckled	1391	1440
Neucleolar	1273	1325
Centomere	1462	1279
Nuclear Membrane	1190	1018
Golgi	362	362
Total	6842	6754

The confusion matrix of our algorithm is

	Prediction						
		homogeneous	speckeled	nucleolar	Centromere	nuclear membrane	golgi
Ground Truth	homogeneous	78.6	10.2	0.2	2.7	4.2	4.1
	speckeled	12.0	81.2	0.5	0.5	1.2	3.6
	nucleolar	1.9	1.7	51.3	1.5	34.2	9.4
	Centromere	2.2	6.7	18.8	70.6	0.0	1.7
	nuclear membrane	10.1	8.1	0.7	12.4	68.7	0.2
	golgi	0.0	3.9	2.6	8.4	4.7	80.4

Table 4.6: Cell level confusion matrix for ICIP 2013

We again compare other dictionary learning algorithms with our algorithm on ICIP 2013 dataset. As in the previous data-sets, here also we keep the pre-processing stage same for all the dictionary learning algorithm. We can see in this data-set also our algorithm is reporting the best accuracy achieved for any dictionary learning algorithm.

Cell Level Accuracy	
Algorithm	accuracy
DL-COPAR [19]	60.1
FDDL [53]	62.8
D-KSVD [20]	68.9
LRSDL [47]	70.7
Our Algorithm	**74.1**

Before concluding results on ICIP 2013 we compare our algorithm on some other standard algorithms published in recent years.

Cell Level Accuracy	
Methods	accuracy (%)
Fisher tensors-based BOW with region co-variance descriptor [11]	70.2
Nokasa et al.[29]	68.8
GoC-LBPs [45]	**75.1**
BoW with gradient orientation histogram and intensity based histogram pooling [43]	**74.4**
BoW with DCT features and spatial decomposition around cell boundary [48]	67.4
Our Algorithm	**74.1**

Amongst the compared algorithms, methods by Nokasa et al.[29] and GoC-LBPs [45]were top two performers of ICPR 2012 competition also. In this datset also our algorithm reported competitive results. The top 3 algorithms are GoC-LBPs [45] and BoW with gradient orientation histogram and intensity based histogram pooling [43] and our algorithm. "BoW with gradient orientation histogram and intensity based histogram pooling" is also a codebook learning algorithm, similar to our proposed algorithm. However, it has much higher computational complexity compared to our algorithm.

4.3 Comparison on SNP data-set

The SNP data-set[49] is another widely accepted data-set for HEp2 cell classification. The data-set was obtained bfrom patients at the "Sullivan Nicolaides Pathology" laboratory, Australia. The data-set has five classes to be classified: "centromere", "coarse speckled", "fine speckled", "homogeneous" and "nucleolar". In total there are 1488 images. The total images are split into two groups 909 and 979 cell images extracted for training and testing respectively. The details of the data-set are as follows

SNP data-set		
class	Test image	train image
Homogeneous	172	188
Coarse Speckled	166	187
Fine Speckled	188	191
Nucleolar	194	188
Centromere	149	183

We compare all the dictionary learning algorithm on SNP datset results in the following table.

Cell Level Accuracy	
Algorithm	accuracy
DL-COPAR [19]	54.1
FDDL [53]	56.2
D-KSVD [20]	59.8
LRSDL [47]	60.9
Our Algorithm	**62.1**

From above table, evidently, our algorithm is the best performer amongst dictionary learning algorithms. However other computationally expensive algorithms have reported good results too. To name a few, William *et.al.* [48] reported an accuracy of 82.5% and Yang *et. al.* [54] reported an accuracy of 80.6% on the SNP data-set.

4.4 Diabetic Retinopathy

Diabetic Retinopathy (DR) is an ailment associated with damages of retinal vascular cells occurring due to long standing diabetes mellitus in patients [25]. (Retinopathy is any damage to the retina of the eyes, which may cause vision impairment. Retinopathy often refers to retinal vascular disease, or damage to the retina caused by abnormal blood flow). In [12] researchers have reported that, Diabetic Retinopathy in recent years has caused blindness and vision impairment in large number of patients, worldwide. They justified their report with figures that, in the year 2016, 0.4 million patients reported to have blindness and 2.6 million other patients are reported to have severe vision impairment due to Diabetic Retinopathy. However, a timely detection of Diabetic Retinopathy helps in early avoidance of the visual impairment and blindness caused by it. However, it is difficult for most patients to get an early treatment as symptoms are not always very strong. Diagnosis of DR is done by close examination of the Fundus image of eye. (Fundus photography involves photographing the rear of an eye; also known as the fundus. Specialized fundus cameras consisting of an intricate microscope attached to a flash enabled camera are used in fundus photograph) But this approach has several drawbacks. The first of

which is that the procedure is time-consuming even for experienced experts. Hence there is a great interest in research community in recent years to effectively design a computer-aided automated diagnosis approache to accurately detect DR efficiently. Many researchers around the world have come up with different approach for efficient detection of diabetic retinopathy in last few years. Doctors (A lesion is any damage or abnormal change in the tissue of an organism, usually caused by disease) try to detect the diabetic retinopathy systems by detect presence of some typical lesions such "hemorrhages", "hard exudates" and "microaneurysms". This idea has been used by many researchers for detection of diabetic retinopathy in fundus images.

4.4.1 Details of The data-set

The data-set used in current article, is re-used from the data-set provided in diabetic retinopathy competition, hosted the website of Kaggle [9]. The data set is originally supplied by EyePACS. It contains 35,126 high resolution fundus photographs taken under different imaging environment. Human experts have labelled fundus images on a scale of 0 to 4 based on the severity of diabetic retinopathy. Table 4.4.1 shows representations of different classes of diabetic retinopathy in the final data-set. The International Clinical Diabetic Retinopathy Scale [22], is defines Referable Diabetic Retinopathy or RDR as the presence of moderate and worse diabetic retinopathy and/or referable diabetic macular edema. Thus, images with labels of 0 and 1 are classified as "without RDR" and relabelled with 0, images with labels of 2, 3 and 4 are classified as "with RDR"' and relabelled with 1. The relabelled data set is reported in Table 4.4.1. However, as in the case of HEp-2 cell, (and other bio medical imaging data-sets), the diabetic retinopathy data-set also suffers from class imbalance. However, the organizers have tried to circumvent this issue. They have paired the data-set in four groups: both eyes are with RDR, both eyes are without RDR, only the left eye is with RDR and only the right eye is with RDR. Then 80% images in each bunch are stored into the training set jointly while the remaining 20% of each bunch will be used as test set, which ensures the proportion of images with different labels is same in both training set and test set. These photographs are captured by different types of cameras in different environment. Due to this reason and many other reasons, unfortunately, there is some noise in the images and labels, which cannot be avoided.

Label	class	number	representation
0	No diabetic retinopathy	25810	73.5%
1	Mild	2443	6.9%
2	Moderate	5292	15.1%
3	severe	873	2.5%
4	Proliferative DR	708	2.0%

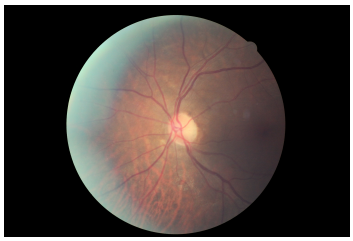

Figure 4.6: No DR

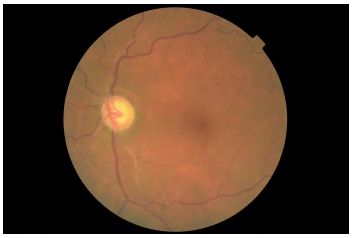

Figure 4.7: Mild

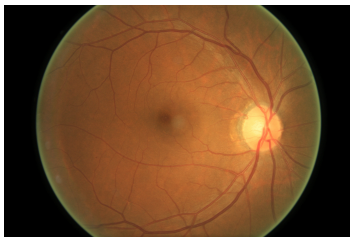

Figure 4.8: Moderate

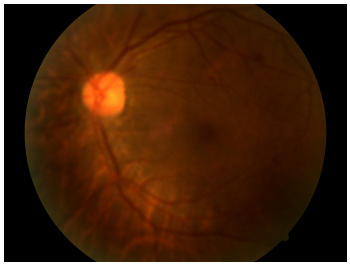

Figure 4.9: Severe

Figure 4.10: Proliferative DR

Label	class	number	representation
0	No RDR	28253	80.4%
1	RDR	6873	19.6%

Figure 4.6 to 4.10 are images for left eye. Corresponding images are available for right eye also.

4.4.2 Results

We compare the results of diabetic retinopathy detection using dictionary learning algorithms. We keep the same SURF based preprocessing step for all the compared algorithms as before.

Fundus image classification	
Algorithm	accuracy
DL-COPAR [19]	52.1
FDDL [53]	51.8
D-KSVD [20]	58.2
LRSDL [47]	58.4
Our Algorithm	**62.1**

Evidently in this case also our algorithm has reported best results among other dictionary learning algorithm. LRSDL closely follows our algorithm in its accuracy. However other dictionary learning algorithms did not report good accuracy in this case. This can be justified that as said in the previous description, the diabetic retinopathy image set considered noises due to camera environment change, image orientations, etc. Also there is noise in the label set. So in general dictionary learning algorithms are not effective in classifying noisy data-sets. However our algorithm has reported good accuracy by improved noise handling.

4.5 Parameter Tuning

We used a few parameters in the algorithm. In this section we shall discuss about the values of the parameters and justification for using those values. In the pre-processing step, SURF is used to extract The features from the images. We have shown in section that overall complexity varies with $number\ of\ atoms$ raised to the power five. However in dictionary learning algorithm uses an over-complete dictionary, hence number of atoms in dictionary is $\geq$ signal dimension. Hence we can say the overall complexity of the algorithm varies with $dimension\ of\ signal$ raised to the power of five. Hence to reduce the computational time, we used 64 dimensional feature descriptor. We used 10 strongest features/key-points by adaptive changing the Hessian Threshold. So we get 640 dimensional signal vector.

In a over-complete systems the number of atoms must be greater than or equal to number of signals. So we take the class-specific dictionary and family specific as well as commonality dictionary atoms all at 720 per class or family as the case may be. Regarding the classifier we have used the SVM as already mentioned. Since we have 720 atoms per class/family. The number of possible output classes is also quite large. So we use linear kernel for the SVM.

For the SVM used in classification step, we used the open source libsvm implementation [8]. For set of L instance label pair points (x_i, y_i), where y_i denotes label and x_i denotes instance. We have the following implementation for the SVM,

$$\min_{w,\xi,b} \qquad \frac{1}{2}w^T w + C\sum_{i=1}^{L}\xi_i$$

subject to

$$y_i(w^T\phi(x_i) + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

We try with different values of parameters and applied between RBF kernel and linear kernel.

- Linear kernel $K(X_i, X_j) = X_i X_j^T$
- RBF Kernel $K(X_i, X_j) = exp(-\gamma \|X_i - X_j\|_2)$

So, as evident from the above description for RBF Kernel we had two parameters (C, γ) and for linear kernel we had only one parameter C. We performed extensive grid search to find the optimum parameter set. We observed that the algorithm obtains highest accuracy with linear kernel. This may be justified as number of attribute (dictionary atoms) is too large for this set. Also for the linear kernel the highest accuracy was obtained for C=4.55 value. The number of possible output classes are six or five depending on the data-set.(For example the ICIP Hep 2 data-set has 6 classes and ICPR 2012 has 5 classes of interest). The number of family or inter-class cluster is dependent on the data-set to be used. We used 3 families all the different data-sets, for ICPR 2012 data-set, for ICIP 2013 data-set, SNP data-set and Diabetic Retinopathy data-set. For ICPR 2012 data-set we have adaptively changed the family number and observed the following graph representation in figure 4.11. The optimum value being obtained by using three families.

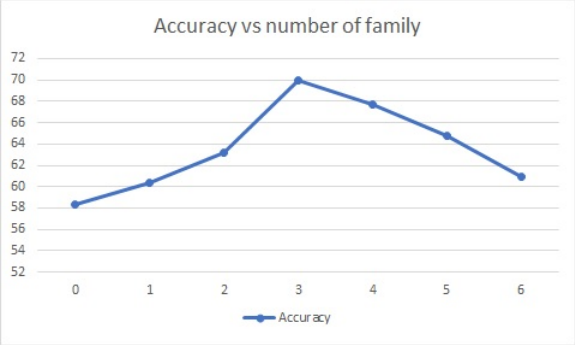

Figure 4.11: Change of accuracy with varying number of family

Similar patterns were observed with other data-sets also. Regarding the initialization of the algorithm using K-SVD we have used 30 iterations of it. We have from equation 3.21 λ_1 is the penalization constant corresponding to the term $\left\|Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0\right\|_F^2$. To observe how strong is the influence or utility is of this penalization term, we vary λ_1 from zero to 1.5 and note down the corresponding accuracy. We obtain a graph as in the following diagram, figure 4.12.

The optimum value for λ_1 is obtained for $\lambda_1 = 1.1$ However the $\lambda_1 = 0$ has very low accuarcy. This shows that the term $\left\|Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0\right\|_F^2$ has good contribution towards the classification problem. Also a relatively flat curve shows that $\left\|Y_i - D_i X_i^i - D_{\mathbb{C}+f} X_i^{\mathbb{C}+f} - D_0 X_i^0\right\|_F^2$ penalization term is a relatively dominant term which is not changed by minor fluctuations. Similarly we have penalization

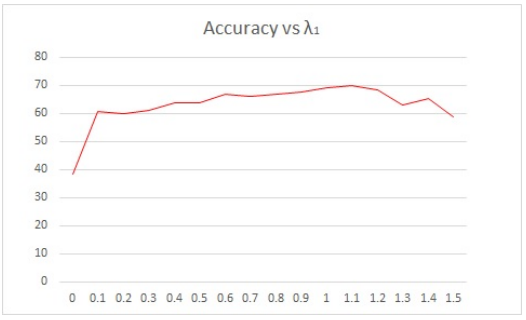

Figure 4.12: Change of accuracy with variation of λ_1

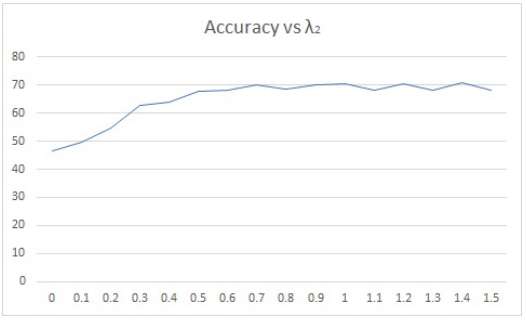

Figure 4.13: Change of accuracy with variation of λ_2

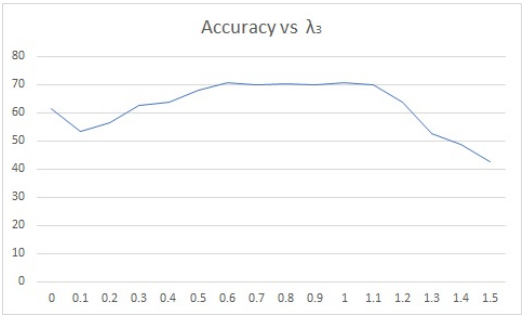

Figure 4.14: Change of accuracy with variation of λ_3

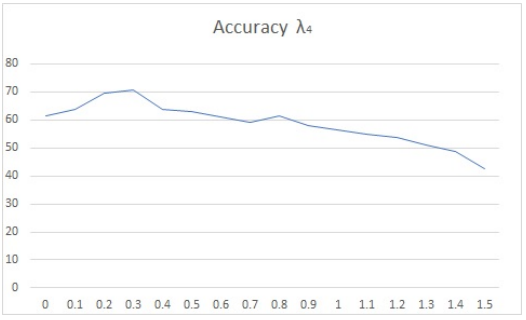

Figure 4.15: Change of accuracy with variation of λ_4

constant λ_2, λ_3, and λ_4. In equation 3.21, λ_2 is the penalization constant corresponding to the term $\left\{\|X_i - M_i\|_F^2 - \|M_i - M_0\|\right\}$. The variation in accuracy with variation in λ_2 is shown in figure 4.13. Evidently threre is increase in accuarcy as λ_2 is increased beyond zero, which justifies the use of this term in our objective function.

Also the optimum accuracy is obtained for $\lambda_2 = 0.8$. Similarly we plot the accuracy versus variation of λ_3 in **??**. . λ_3 is the penalization corresponding to the term $\sum_{j=0,\, j\neq i}^{\mathbb{C}+\mathbb{F}} \left\|D_i^T D_j\right\|_F^2$. This term makes the dictionary more discriminative, and loses the between class features. So it results in a drop of accuarcy, as there are many inter-class similarity based features in our data-set.

λ_4 is the penalization corresponding to $\|X\|_1$. We obtain similar graph as shown in figure 4.15. A high value in λ_4 leads to high sparsity penalization. So we obtain very low non zero significant values in X matrix. So, increasing λ_4 increases sparsity unnecessarily and results in decrease in accuarcy.

Chapter 5

Conclusion and scope of future work

Bio-medical image processing and pattern recognition majorly suffers from high in-class variation and low inter-class variation, both of which is undesired for effective classification. We tried to circumvent these issues with a modification to dictionary learning approach. Medical imaging data-set also suffers from class imbalance. we tried to avoid the bias due to data imbalance by using a SVM classifier on a novel setup. Results showed that our algorithm is superior to other dictionary algorithm methods and competitive to winners of different competitions.

However we considered one class can only belong to one family at a time for computational simplicity. However it is a topic of further research to check whether considering one class belonging to multiple classes helps in obtaining better result. Also we only used dependencies between classes and that has been considered as family. But in future extension of this work we may consider dependencies between families. In more generalised case a graph based structure can be considered where each node in the graph is a sub-dictionary. In such a graph based configuration we can better model the dependencies between the sub-dictionaries. Moreover we have only considered discriminative relationship between family specific dictionaries. But threre may be cooperative relationship between some family specific dictionaries and discriminative relationship between some family specific dictionaries. The existing structure does not allow such dependencies to be considered. However it may be a research to design a structure that may incorporate such dependencies.

Appendix A

Deriving relevant matrix calculus formulae

To get the optimal values of different dictionaries we have to solve the objective function. We will be using several matrix calculus results. I briefly derive them in the following section. The derivative of a scalar f with respect to a matrix $A \in \mathbb{R}^{\mathbb{M}\times\mathbb{N}}$ can be written as

$$\begin{bmatrix} \frac{\partial f}{\partial A_{11}} & \frac{\partial f}{\partial A_{12}} & \cdots & \frac{\partial f}{\partial A_{N1}} \\ \frac{\partial f}{\partial A_{21}} & \frac{\partial f}{\partial X_{22}} & \cdots & \frac{\partial f}{\partial A_{N1}} \\ \cdots & \cdots & & \cdots \\ \frac{\partial f}{\partial A_{M1}} & \frac{\partial f}{\partial A_{M2}} & \cdots & \frac{\partial f}{\partial A_{MN}} \end{bmatrix}$$

WE decompose matrix multiplication to index-based scalar multiplication, $[AB]_{ik} = \sum_j A_{ij}B_{jk}$ and similarly the matrix product and the matrix product ABC^T has elements:

$$[ABC^T]_{il} = \sum_j A_{ij}[BC^T]_{jl} = \sum_j A_{ij} \sum_k B_{jk}C_{lk} = \sum_j \sum_k A_{ij}B_{jk}C_{lk}$$

Next using this indexing concepts we derive some first order derivatives. Let

$$f = trace[ANB]$$

Using index notations as shown above we can write as

$$f = \sum_i [ANB]_{ii} = \sum_i \sum_j A_{ij}[NB]_{ji} = \sum_i \sum_j A_{ij} \sum_k N_{jk}B_{ki} = \sum_i \sum_j \sum_k A_{ij}N_{jk}B_{kl}$$

Now taking the partial derivative with respect to N_{jk} we do have

$$\frac{\partial f}{\partial N_{jk}} = \sum_i A_{ij}B_{ki} = [BA]_{kj}$$

Now we express this index based notation to matrix multiplication form again

$$\frac{\partial\ trace[ANB]}{\partial N} = A^T B^T \tag{A.1}$$

Similarly, we have:

$$f = trace[AN^T B] = \sum_i \sum_j \sum_k A_{ij} N_{kj} B_{kl}$$

so that the derivative is:

$$\frac{\partial f}{\partial N_{kj}} = \sum_i A_{ij} B_{ki} = [BA]_{kj}$$

Thus, we have:

$$\frac{\partial\ trace[AN^T B]}{\partial N} = BA \tag{A.2}$$

Multiple order of matrix derivatives can also be derived as follows

$$f = trace[ANBNC^T] = \sum_i \sum_j \sum_k \sum_l \sum_m A_{ij} N_{jk} B_{kl} N_{lm} C_{im}$$

The derivative has contributions from both appearances of N In index notation:

$$\frac{\partial f}{\partial N_{jk}} = \sum_i \sum_l \sum_m A_{ij} B_{kl} N_{lm} C_{im} = [BNC^T A]_{kj}$$

$$\frac{\partial f}{\partial N_{lm}} = \sum_i \sum_j \sum_k A_{ij} N_{jk} B_{kl} C_{lm} = [C^T AXB]_{ml}$$

Transposing appropriately and summing the terms together, we have:

$$\frac{\partial\ trace[ANBNC^T]}{\partial N} = \frac{\partial\ trace[ANP]}{\partial N} + \frac{\partial\ trace[QNC^T]}{\partial N} = A^T P^T + Q^T C$$

where $P = BNC^T$ and $Q = ANB$ So we separately evaluated the matrix derivative for each appearance of N assuming that everything else constant (including other N's). We utilize the results derived above to evaluate partial derivative of frobenius Norm

$$f = \|N - WH\|_F^2 = trace\left[\Big(N - WH\Big)\Big(N - WH\Big)^T\right] = \sum_i \sum_k \left(N_{ik} - \sum_j W_{ij} H_{jk}\right)^2$$

We can work with the expression in index notation, but it's easier to work directly with matrices and apply the results derived earlier. Suppose we want to find the derivative with respect to W. Expanding the matrix outer product, we have:

$$f = trace[NN^T] - trace[NH^TW^T] - trace[WHN^T] + trace[WHH^TW^T]$$

Applying equation A.1 and equation A.2 we easily deduce that

$$\frac{\partial \ trace\Big[(N - WH)(N - WH)^T\Big]}{\partial W} = -2NH^T + 2WHH^T \tag{A.3}$$

this can be restated as

$$\frac{\partial \ \|N - WH\|_F^2}{\partial W} = -2NH^T + 2WHH^T \tag{A.4}$$

similarly we have

$$\frac{\partial \ trace\Big[(N - WH)(N - WH)^T\Big]}{\partial H} = -2W^TN + 2WW^TH \tag{A.5}$$

this equation can also be restated as

$$\frac{\partial \ \|N - WH\|_F^2}{\partial H} = -2W^TN + 2WW^TH \tag{A.6}$$

Bibliography

[1] Aharon, M., Elad, M., Bruckstein, A., et al.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on signal processing 54(11), 4311 (2006)

[2] Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis & Machine Intelligence (12), 2037–2041 (2006)

[3] Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: European conference on computer vision. pp. 404–417. Springer (2006)

[4] Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM journal on imaging sciences 2(1), 183–202 (2009)

[5] Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: Proceedings of the 6th ACM international conference on Image and video retrieval. pp. 401–408. ACM (2007)

[6] Bryt, O., Elad, M.: Compression of facial images using the k-svd algorithm. Journal of Visual Communication and Image Representation 19(4), 270–282 (2008)

[7] Cai, T.T., Wang, L.: Orthogonal matching pursuit for sparse signal recovery with noise. Institute of Electrical and Electronics Engineers (2011)

[8] Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011), software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

[9] https://www.kaggle.com/c/diabeticretinopathy detection/: Kaggle. (jul. 27, 2015). diabetic retinopathy detection. accessed: Jun 20, 2019. [Online]. Available: 39, 178–193 (2017)

[10] Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image processing 15(12), 3736–3745 (2006)

[11] Faraki, M., Harandi, M.T., Wiliem, A., Lovell, B.C.: Fisher tensors for classifying human epithelial cells. Pattern Recognition 47(7), 2348–2359 (2014)

[12] Flaxman, S.R., Bourne, R.R., Resnikoff, S., Ackland, P., Braithwaite, T., Cicinelli, M.V., Das, A., Jonas, J.B., Keeffe, J., Kempen, J.H., et al.: Global causes of blindness and distance vision impairment 1990–2020: a systematic review and meta-analysis. The Lancet Global Health 5(12), e1221–e1234 (2017)

[13] Foggia, P., Percannella, G., Soda, P., Vento, M.: Early experiences in mitotic cells recognition on hep-2 slides. In: 2010 IEEE 23rd International Symposium on Computer-Based Medical Systems (CBMS). pp. 38–43. IEEE (2010)

[14] Foggia, P., Percannella, G., Soda, P., Vento, M.: Benchmarking hep-2 cells classification methods. IEEE transactions on medical imaging 32(10), 1878–1889 (2013)

[15] Gabor, D.: Theory of communication. part 1: The analysis of information. Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering 93(26), 429–441 (1946)

[16] Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE transactions on image processing 19(6), 1657–1663 (2010)

[17] Haralick, R.M., Shanmugam, K., et al.: Textural features for image classification. IEEE Transactions on systems, man, and cybernetics (6), 610–621 (1973)

[18] Kastaniotis, D., Theodorakopoulos, I., Economou, G., Fotopoulos, S.: Hep-2 cells classification using locally aggregated features mapped in the dissimilarity space. In: 13th IEEE International Conference on BioInformatics and BioEngineering. pp. 1–4. IEEE (2013)

[19] Kong, S., Wang, D.: A dictionary learning approach for classification: separating the particularity and the commonality. In: European conference on computer vision. pp. 186–199. Springer (2012)

[20] Kong, X., Li, K., Cao, J., Yang, Q., Wenyin, L.: Hep-2 cell pattern classification with discriminative dictionary learning. Pattern Recognition 47(7), 2379–2388 (2014)

[21] Larsen, A.B.L., Vestergaard, J.S., Larsen, R.: Hep-2 cell classification using shape index histograms with donut-shaped spatial pooling. IEEE transactions on medical imaging 33(7), 1573–1580 (2014)

[22] Levels, E.: International clinical diabetic retinopathy disease severity scale detailed table (2002)

[23] Li, K., Yin, J., Lu, Z., Kong, X., Zhang, R., Liu, W.: Multiclass boosting svm using different texture features in hep-2 cell staining pattern classification. In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012). pp. 170–173. IEEE (2012)

[24] Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision 60(2), 91–110 (2004)

[25] Luu, C.D., Szental, J.A., Lee, S.Y., Lavanya, R., Wong, T.Y.: Correlation between retinal oscillatory potentials and retinal vascular caliber in type 2 diabetes. Investigative ophthalmology & visual science 51(1), 482–486 (2010)

[26] Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proceedings of the 26th annual international conference on machine learning. pp. 689–696. ACM (2009)

[27] Manivannan, S., Li, W., Akbar, S., Wang, R., Zhang, J., McKenna, S.J.: Hep-2 cell classification using multi-resolution local patterns and ensemble svms. In: 2014 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images. pp. 37–40. Ieee (2014)

[28] Meroni, P.L., Schur, P.H.: Ana screening: an old test with new recommendations. Annals of the rheumatic diseases 69(8), 1420–1422 (2010)

[29] Nosaka, R., Fukui, K.: Hep-2 cell classification using rotation invariant co-occurrence among local binary patterns. Pattern Recognition 47(7), 2428–2436 (2014)

[30] Nosaka, R., Ohkawa, Y., Fukui, K.: Feature extraction based on co-occurrence of adjacent local binary patterns. In: Pacific-Rim Symposium on Image and Video Technology. pp. 82–91. Springer (2011)

[31] Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern recognition 29(1), 51–59 (1996)

[32] Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis & Machine Intelligence (7), 971–987 (2002)

[33] Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis & Machine Intelligence (7), 971–987 (2002)

[34] Olshausen, B.A., Field, D.J.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381(6583), 607 (1996)

[35] Olshausen, B.A., Field, D.J.: Sparse coding with an overcomplete basis set: A strategy employed by v1? Vision research 37(23), 3311–3325 (1997)

[36] Pietikäinen, M., Hadid, A., Zhao, G., Ahonen, T.: Computer vision using local binary patterns, vol. 40. Springer Science & Business Media (2011)

[37] Ponomarev, G.V., Arlazarov, V.L., Gelfand, M.S., Kazanov, M.D.: Ana hep-2 cells image classification using number, size, shape and localization of targeted cell regions. Pattern Recognition 47(7), 2360–2366 (2014)

[38] Qi, X., Xiao, R., Li, C.G., Qiao, Y., Guo, J., Tang, X.: Pairwise rotation invariant co-occurrence local binary pattern. IEEE transactions on pattern analysis and machine intelligence 36(11), 2199–2213 (2014)

[39] Qi, X., Xiao, R., Li, C.G., Qiao, Y., Guo, J., Tang, X.: Pairwise rotation invariant co-occurrence local binary pattern. IEEE transactions on pattern analysis and machine intelligence 36(11), 2199–2213 (2014)

[40] Qi, X., Zhao, G., Li, C.G., Guo, J., Pietikäinen, M.: Hep-2 cell classification via combining multiresolution co-occurrence texture and large region shape information. IEEE journal of biomedical and health informatics 21(2), 429–440 (2015)

[41] Qi, X., Zhao, G., Li, C.G., Guo, J., Pietikäinen, M.: Hep-2 cell classification via fusing texture and shape information. arXiv preprint arXiv:1502.04658 (2015)

[42] Ramirez, I., Sprechmann, P., Sapiro, G.: Classification and clustering via dictionary learning with structured incoherence and shared features. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 3501–3508. IEEE (2010)

[43] Shen, L., Lin, J., Wu, S., Yu, S.: Hep-2 image classification using intensity order pooling based features and bag of words. Pattern Recognition 47(7), 2419–2427 (2014)

[44] Theodorakopoulos, I., Kastaniotis, D., Economou, G., Fotopoulos, S.: Hep-2 cells classification via fusion of morphological and textural features. In: 2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE). pp. 689–694. IEEE (2012)

[45] Theodorakopoulos, I., Kastaniotis, D., Economou, G., Fotopoulos, S.: Hep-2 cells classification via sparse representation of textural features fused into dissimilarity space. Pattern Recognition 47(7), 2367–2378 (2014)

[46] Varma, M., Zisserman, A.: A statistical approach to texture classification from single images. International journal of computer vision 62(1-2), 61–81 (2005)

[47] Vu, T.H., Monga, V.: Fast low-rank shared dictionary learning for image classification. IEEE Transactions on Image Processing 26(11), 5160–5175 (2017)

[48] Wiliem, A., Sanderson, C., Wong, Y., Hobson, P., Minchin, R.F., Lovell, B.C.: Automatic classification of human epithelial type 2 cell indirect immunofluorescence images using cell pyramid matching. Pattern Recognition 47(7), 2315–2324 (2014)

[49] Wiliem, A., Wong, Y., Sanderson, C., Hobson, P., Chen, S., Lovell, B.C.: Classification of human epithelial type 2 cell indirect immunofluoresence images via codebook based descriptors. In: IEEE Workshop on Applications of Computer Vision (WACV) (2013)

[50] Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE transactions on pattern analysis and machine intelligence 31(2), 210–227 (2008)

[51] Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE transactions on image processing 19(11), 2861–2873 (2010)

[52] Yang, J., Yu, K., Gong, Y., Huang, T.S., et al.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR. vol. 1, p. 6 (2009)

[53] Yang, M., Zhang, L., Feng, X., Zhang, D.: Fisher discrimination dictionary learning for sparse representation. In: 2011 International Conference on Computer Vision. pp. 543–550. IEEE (2011)

[54] Yang, Y., Wiliem, A., Alavi, A., Hobson, P.: Classification of human epithelial type 2 cell images using independent component analysis. In: 2013 IEEE International Conference on Image Processing. pp. 733–737. IEEE (2013)

[55] Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Transactions on Pattern Analysis & Machine Intelligence (6), 915–928 (2007)