Abstract—Microscopy image classification is important in various biomedical applications, such as cancer subtype identification and protein localization for high content screening. To achieve automated and effective microscopy image classification, the representative and discriminative capability of image feature descriptors is essential. To this end, in this study we propose a new feature representation algorithm to facilitate automated microscopy image classification. In particular, we incorporate Fisher vector (FV) encoding with multiple types of local features that are handcrafted or learned, and we design a separation-guided dimension reduction (SDR) method to reduce the descriptor dimension while increasing its discriminative capability. Our method is evaluated on four publicly available microscopy image datasets of different imaging types and applications, including the UCSC breast cancer dataset, MICCAI 2015 CBTC challenge dataset, and IICBU malignant lymphoma and RNAi datasets. Our experimental results demonstrate the advantage of the proposed low-dimensional FV representation, showing consistent performance improvement over the existing state-of-the-art and the commonly used dimension reduction techniques.

Index Terms—Fisher vector, feature learning, dimensionality reduction, discriminative learning.

I. INTRODUCTION

Visual analysis of microscopy images is important in many biomedical applications. For example, the final diagnosis of cancers for staging and grading is regularly performed by the histopathology examination of biopsy tissue samples. As another example, the localization of protein expressions within a cell helps to facilitate high content screening of cellular phenotypes in biological and drug development studies. In both examples, the visual analysis is essentially an image classification problem. Specifically, for cancer analysis, the problem can be the classification of tissue samples as benign / malignant (Fig. 1a), or the classification of various cancer subtypes (Figs. 1b and 1c). Also, the localization of protein expressions can be approached by classifying the subcellular patterns in images generated from knockdown of certain genes (Fig. 1d).

Computerized microscopy image classification approaches have been developed over the years with the aim to provide efficient and consistent image interpretation automatically. In the domain of digital pathology, approaches have been designed for grading or subtyping of various cancers including squamous cell carcinoma [1], prostate cancer [2], brain tumor [3], and breast cancer [4]. In these approaches, histological biomarkers are detected and quantified to encode the morphological characteristics that are critical for the histopathological analysis. For example, to determine the aggressiveness of breast cancer, the ratio of tubule nuclei to overall number of nuclei is an important biomarker and a deep learning technique is designed to detect the tubule nuclei [4]. More reviews are provided in [5] as well. The advantage of these approaches is that the defined biomarker is biologically inspired and the quantification could directly correlate with the histopathological criteria used by the pathologists. However, domain-specific knowledge is required to design such methods, and in some cases extensive ground truth labeling is necessary to facilitate a learning-based algorithm.

On the other hand, there are also methods that are developed to perform image-level classification without detecting or quantifying a particular biomarker and such methods are typically evaluated on a variety of different applications [6], [7], [8], [9], [10], [11], [12], [13], [14]. For example, in one study [7], a customized feature set is designed and evaluated on nine different classification tasks, including tissue age differentiation, subcellular protein localization, lymphoma subtyping, and pollen grain distinction. In another study [11], an image classification platform is developed for a standardized support of different problems, such as the quantification of subcellular objects, cell phenotype recognition, and histopathological classification. Although such general methods are usually less effective than approaches incorporating domain specific designs, the generalization property can be particularly useful to biomedical researchers working on new studies, when there are no customized imaging analysis tools available. Our aim of this study belongs to this type, i.e. to design an image-level classification method for varying microscopy imaging applications without detecting biomarkers.

The key component in automated microscopy image clas-
sification, regardless of the clinical application, is the feature representation of the images. Features that are representative and discriminative are fundamental to the classification performance in applications such as histopathological classification [2], [12], [13], [3], [15], [16], [17], [18], [19], [20], cell detection / segmentation [21], [22], [23], [24], [25], event detection [26], [27], [28], and subcellular localization [7], [11], [29]. In particular, microscopy images of different classes often exhibit high level of visual similarities while images of the same class often show varying patterns that are difficult to summarize. Imaging artifacts also introduce noise. These factors make it challenging to design an effective feature representation for microscopy images. Existing studies often use a combination of standard feature descriptors such as Haralick textures, filter banks, scale-invariant feature transform (SIFT), local binary patterns (LBP) and histogram of oriented gradients (HOG) [7], [11], [29], [12], [30], [14], [31]. Customized feature design has also been conducted to achieve better performance [26], [2], [13], [3], [32], [33], [15]. In addition, recent studies have demonstrated promising performance using automated feature learning based on unsupervised models such as autoencoder and its variations [21], [16], [22], [17], [18], [34], [20], [35]; and supervised models such as the convolutional neural network (CNN) [27], [28], [23], [24], [25].

Among these features, a relatively studied feature descriptor for microscopy image analysis is the Fisher vector (FV) [36]. FV encodes patch-level local features into a global description of the whole image by first constructing a Gaussian mixture model (GMM) from the local features with unsupervised learning, then concatenating the differences between local features and the GMM centers as the FV descriptor. This model has demonstrated successful uses in a variety of general imaging applications such as image classification [37], face recognition [38], object detection [39], and texture characterization [40], [41]. While dense SIFT is typically used as the local features, different patch-level local features have been used to better represent microscopy images. In particular, local features learned using deconvolution network are encoded into FV descriptors, which are used to identify subtypes of ovarian carcinomas [16]; and FV descriptors with LBP-like local features are used to classify HEp-2 cells [33]. Large performance improvement over standard handcrafted features has been demonstrated in both studies.

FV encoding can also be integrated with CNN by using a CNN model to extract the local features. Such an approach demonstrates significant improvement for texture classification over using deep learning alone or the standard FV encoding of dense SIFT local features [42]. In addition, while the CNN model can be trained from scratch on the target dataset, to reduce the method complexity, a transfer learning approach with a CNN model pretrained on the ImageNet database can also be used. Various biomedical imaging studies have shown success in using pretrained CNN models (without FV encoding), and suggested that biomedical images indeed share similar low-level features with natural images even though the high-level semantics make them appear very different [43], [44], [45].

Besides the good discriminative power, FV descriptors also have much higher dimensions than the usual handcrafted or learned descriptors. The linear-kernel support vector machine (SVM) is thus regularly used with FV descriptors for the classification task. SVM is intrinsically capable of handling high-dimensional data and the linear kernel is computationally efficient. However, in microscopy image studies, the image dataset is typically much smaller than the general imaging datasets. With the limited amount of images, the unsupervised learning of FV descriptors and supervised learning of SVM classifier could overfit to the training data and the resultant model might not generalize well to unseen cases. Therefore, it is intuitive to consider that reducing the dimension of FV descriptors could help to highlight only the critical feature variables and improve the feature representativeness. It has been shown that dimensionality reduction of FV descriptors based on large margin distance metric learning helps to improve the results of face recognition [38], [46].

In this study, we explore the use of FV descriptors and dimensionality reduction algorithm for microscopy image classification. We construct FV descriptors with three types of local features: dense SIFT, deep belief networks (DBN) [47] that is learned in an unsupervised manner, and a CNN model...
[48] that is pretrained on ImageNet. We choose these three local features as exemplars of the handcrafted, unsupervised learning, and supervised learning descriptors. To the best of our knowledge, FV encoding of DBN or CNN local features has not been investigated in microscopy image studies. We also design a new dimensionality reduction algorithm, namely the separation-guided dimension reduction (SDR), which helps to increase the feature space separation between different classes and reduce the separation within the same class based on an integrated learning of distance metric and discriminative classification. In addition, we design our method with minimum domain-specific knowledge of imaging characteristics so that our method can be commonly applied to various microscopy imaging applications. Fig. 2 illustrates the overall flow of our proposed method.

Our methodological contributions are summarized as follows. (i) We find that with FV encoding, the CNN-based local features are generally more effective than dense SIFT and DBN features; and combining the FV descriptors of different local features normally leads to higher classification performance. (ii) We show that with our proposed SDR algorithm, the dimension reduced descriptor is more discriminative than the high-dimensional FV descriptor and the advantage is especially prominent on small datasets. (iii) Our results show that SDR outperforms the commonly used dimensionality reduction techniques, such as the principal component analysis (PCA), linear discriminant analysis (LDA), generalized discriminant analysis (GDA) [49], ISOMAP [50], and full matrix learning (FML) [38]. (iv) The effectiveness of our method is demonstrated on four public datasets of different clinical applications, including the UCSC breast cancer dataset for differentiation between benign and malignant tumors [13], MICCAI 2015 CBTC challenge dataset for classification of two brain tumor types, IICBU malignant lymphoma dataset for distinguishing three malignant lymphoma types, and IICBU RNAi dataset for classifying ten gene phenotypes [51]. We achieve improved performance over the state-of-the-art approaches reported on these datasets.

A preliminary version of this work has been published in the conference [52]. Compared to the preliminary study [52], an enhanced dimensionality reduction algorithm (SDR) is presented in this paper, which integrates the distance metric with discriminative classification for model optimization and contains a revised technique for training data creation. Also, CNN-based local features are incorporated in addition to the dense SIFT and DBN features. We have also performed more thorough performance evaluation, and included the IICBU RNAi dataset to demonstrate our method performance on a different type of microscopy imaging.

II. FISHER VECTOR REPRESENTATION

FV [36] is a feature encoding technique that aggregates a dense set of local features into a high-dimensional descriptor that represents the image-level characteristics. The underlying algorithm is based on the Fisher kernel [53]. Briefly, the generation process of the local features can be modeled by a probability density function \( u \). Based on \( u \), the gradient of the log-likelihood of the local features can be computed to represent the set of local features. This gradient vector is the main component in computing the FV descriptor, and \( u \) is assumed to follow a GMM model to facilitate an analytical approximation of the Fisher information.

To compute the FV descriptor, a GMM with \( K \) components is first generated based on a set of local features extracted from the training images. Each Gaussian component is represented by its mean vector \( \mu_k \), standard deviation vector \( \sigma_k \), and mixture weight \( \nu_k \). For a test image with \( N \) dense local features, each local feature \( f_n \) is soft assigned to each of the Gaussian components with the assignment weight computed as:

\[
\rho_n(k) = \frac{N(f_n | \mu_k, \sigma_k^2)\nu_k}{\sum_{j=1}^{K} N(f_n | \mu_j, \sigma_j^2)\nu_j}
\]

where \( N(\cdot) \) is the Gaussian density function. The average first and second order differences between the local features and each GMM center are then computed as:

\[
\tau_k^{(1)} = \frac{1}{N\sqrt{\nu_k}} \sum_{n=1}^{N} \rho_n(k) (f_n - \mu_k) / \sigma_k,
\]

\[
\tau_k^{(2)} = \frac{1}{N\sqrt{2\nu_k}} \sum_{n=1}^{N} \rho_n(k) [(f_n - \mu_k)^2 / \nu_k - 1].
\]

The FV descriptor of the test image is the concatenation of all first and second order differences respective to the \( K \) components: \( x = [\tau_1^{(1)}, \tau_1^{(2)}, \ldots, \tau_K^{(1)}, \tau_K^{(2)}] \), which is also power and L2 normalized.

The FV descriptor \( x \) has a dimension of \( 2KD \), where \( D \) is the dimension of the local features. The parameter \( K \), which is the number of Gaussian components, is set to 64. While typically larger \( K \) values would increase the discriminative power of FV descriptors [36], \( K = 64 \) is the usual setting used in texture classification [40], [41], [42] and we find that the value also provides good classification performance for our problems. A larger \( K \) value (e.g., 80 or 128) would improve the classification performance slightly but impose higher computational cost.

Various types of local features can be integrated with FV encoding. In our design, we employ the following three local features.

1) **SIFT**: SIFT is the standard local feature used with FV encoding. A multi-scale dense feature extraction is performed, with spatial bins of 4, 6, 8, 10, and 12 pixels and sampling of every two pixels. Following the standard setup, the 128-dimensional SIFT feature is reduced to \( D = 64 \) dimensions using PCA before FV encoding. The resultant SIFT-based FV descriptor is thus \( 2KD = 2 \times 64 \times 64 = 8192 \) dimensional.

2) **DBN**: While SIFT is a highly effective yet handcrafted feature, unsupervised feature learning has become increasingly popular. In particular, DBN, which consists of multiple layers of restricted Boltzmann machines (RBMs), has been successfully applied in recent biomedical imaging studies [54], [55]. To apply DBN to our data, we construct a two-layer DBN with each layer having 64 output neurons. A deeper DBN is not used due to the limited amount of training data. The model is trained on half-overlapping image patches of 8 \times 8 pixels.
For a test image, the learned model is applied to each patch, and the 64 features from each layer are concatenated to form the local DBN feature. Similar to SIFT, PCA is also applied to reduce the feature dimension to 64. The resultant DBM-based FV descriptor is thus 2 × 64 × 64 = 8192 dimensional. Note that we use a small patch size so that a relatively small number of neurons can be used and subsequently a small feature dimension is produced. Our experiments show that a higher feature dimension would result in a small degree of reduction in the classification performance.

3) CNN: CNN is a supervised feature learning technique, which is a multilayer neural network including convolutional and fully connected layers. It has been widely incorporated into biomedical imaging and has often demonstrated good performance [23], [24], [25]. Models that are pretrained on ImageNet have also been applied in biomedical imaging showing good results, implying that biomedical and natural images indeed exhibit similar visual characteristics at the low level [43], [44], [45]. In this study, the VGG-VD model (very deep with 19 layers) pretrained on ImageNet [48] is applied to the image. The image is rescaled to multiple sizes with scales of 2^s, s = −3, −2.5, . . . , 1.5, and 512-dimensional local features are densely extracted from the last convolutional layer. These local features are then encoded using FV. The resultant CNN-based FV descriptor is of 2 × 64 × 512 = 65536 dimensions. Note that unlike SIFT and DBN, PCA is not applied to reduce the dimension of the CNN local features, since that would largely reduce the classification performance as observed in our empirical study.

Note that when applying CNN in biomedical imaging, it is more common to train a new CNN model from the biomedical images than using the pretrained models. However, due to the small number of biomedical images available, typically patch-based processing and customized CNN architectures are required. Since our focus of this study is not about customizing the CNN design, we only experimented with simple training approaches by varying the AlexNet [56] architecture. We find that the pretrained VGG-VD model actually provides better classification performance, and the basic fine-tuning process (replacing the last layer in VGG-VD with the actual number of classes and performing backpropagation to a certain layer) decreases the classification performance as well. Therefore, we have adopted the pretrained VGG-VD model and this approach also means that our method can be easily applied to different applications. On the other hand, we also suggest that it would be possible to design a more effective patch-based CNN approach and use this customized CNN model as the local feature extractor.

We also note that besides these three local features, other types of features can also be used, as long as they provide dense patch-level feature descriptions. FV encoding will work with such local features in the same way.

III. SEPARATION-GUIDED DIMENSION REDUCTION

In this section, we describe our SDR method that reduces the feature dimension of FV descriptors for SVM classification. Formally, given a test image with a d-dimensional FV descriptor \( x \in \mathbb{R}^d \), \( d \in \{8192, 65536\} \), we perform a binary classification using the linear-kernel SVM by:

\[
y = \text{sign}(\omega \cdot \phi(x) + b)
\]

where \( y \in \{-1, 1\} \) indicates the class label, and \( \phi(x) \in \mathbb{R}^h \) is the dimension reduced descriptor of \( h \) dimensions \( (h \ll d) \). Following the standard SVM definition, \( \omega \in \mathbb{R}^h \) is the weight vector and \( b \) is the bias value. The dimension reduced descriptor \( \phi(x) \) is computed as:

\[
\phi(x) = Mx
\]

where \( M \in \mathbb{R}^{h \times d} \) is the projection matrix that is learned along with the classification objective (see Section III.A). The low-dimensional descriptor \( \phi(x) \) is expected to provide higher discriminative power and lead to better classification result.
A. Learning of Model Parameters

We formulate a discriminative model to derive the parameters $\omega$, $b$, and $M$. The model integrates dimension reduction and classification with an objective of maximizing the separation between different classes. Specifically, assume that a set of $I$ training samples are given: $\{x_i, y_i : i = 1, \ldots, I\}$. The following optimization objective is defined:

$$\arg\min_{\omega} \frac{1}{2}\|\omega\|^2 + C \sum_{i=1}^{I} \xi_i,$$

s.t. $y_i(\omega \cdot \phi(x_i) + b) \geq 1 - \xi_i$, $\xi_i \geq 0$, $\forall i$, \hspace{1cm} (6)

$$\|Mx_i - Mx_j\|^2 > g, \forall y_i \neq y_j,$$

$$\|Mx_i - Mx_j\|^2 < g, \forall y_i = y_j.$$ \hspace{1cm} (8)

Eqs. (6) and (7) are the standard SVM formulation for optimizing the parameters $\omega$ and $b$ with $\xi_i$ as the slack variable and $C$ as the constant scalar. Eq. (8) is an additional constraint for the projection matrix $M$ based on pairwise distances. This distance metric constraint specifies that we expect large separation (i.e. an Euclidean distance larger than a threshold $g$) between descriptors $(x_i$ and $x_j$, $j = 1, \ldots, I)$ of different classes ($y_i \neq y_j$), and small separation between descriptors of the same class. In other words, this constraint encourages larger inter-class distinction and lower intra-class variation in the feature space. Subsequently, by transforming the descriptors $x_i$ into lower dimensions $\phi(x_i)$, it would be easier to establish a separation hyperplane by SVM and hence more accurate classification would be expected.

To solve this optimization problem, we note that with $\phi(x_i)$ in Eq. (7), $M$ can be considered as a latent variable in the SVM model. The distance metric in Eq. (8) is thus integrated with the discriminative classification of SVM. We design an alternative optimization approach to derive $\omega$ and $b$, and $M$ and $g$.

First, assume $M$ is given. The low-dimensional descriptors $\phi(x_i)$ are then computed, and $\omega$ and $b$ are derived using the standard linear SVM model defined in Eqs. (6) and (7).

Second, with known $\omega$ and $b$, $M$ and $g$ are derived based on the constraints in Eqs. (7) and (8), which are rewritten as:

$$\theta_{ij}(g - \|Mx_i - Mx_j\|^2) +$$

$$\alpha\{y_i\omega \cdot (Mx_i) + b\} + y_j\omega \cdot (Mx_j) + b\} > 1, \hspace{1cm} (9)$$

where $\theta_{ij}$ is $1$ if $y_i = y_j$ and $-1$ otherwise, and $\alpha$ is a constant. With hinge loss, the optimization function is formulated as:

$$\arg\min_{M, g} \sum_{i,j} \max(1 - s_{ij}, 0), \hspace{1cm} (10)$$

where

$$s_{ij} = \theta_{ij}(g - (x_i - x_j)^TM^TM(x_i - x_j)) +$$

$$\alpha\{y_i\omega^TMx_i + b\} + y_j\omega^TMx_j + b\}, \hspace{1cm} (11)$$

This $s_{ij}$ helps to encourage two objectives: (i) $x_i$ and $x_j$ should be classified correctly as $y_i$ and $y_j$; and (ii) when $x_i$ and $x_j$ are from the same class (i.e. $\theta_{ij} = 1$), the distance between $x_i$ and $x_j$ should be less than $g$, and when $x_i$ and $x_j$ are from different classes (i.e. $\theta_{ij} = -1$), the distance between $x_i$ and $x_j$ should be greater than $g$.

A stochastic sub-gradient method is then used to compute $M$ and $g$. In particular, at each iteration $t$, a pair of training data $x_i$ and $x_j$ is sampled and used to update $M$ by:

$$M_{t+1} = \begin{cases} M_t & \text{if } s_{ij} > 1 \\ M_t - \gamma\Delta_{ij} & \text{otherwise} \end{cases} \hspace{1cm} (12)$$

where $\gamma$ is a constant learning rate, and

$$\Delta_{ij} = \theta_{ij}M_t(x_i - x_j)M_t(x_i - x_j)^T - \alpha(y_i\omega^T + y_j\omega^T). \hspace{1cm} (13)$$

The threshold $g$ is also updated by:

$$g_{t+1} = \begin{cases} g_t & \text{if } s_{ij} > 1 \\ g_t + \lambda s_{ij} & \text{otherwise} \end{cases} \hspace{1cm} (14)$$

where $\lambda$ is the learning rate. The iteration continues until convergence or the maximum number of iterations is reached. Note that the objective function Eq. (10) is not convex in $M$, hence initialization is important to obtain a good solution (described in Step 1 below). Also, our experiments show that the classification performance is not sensitive to the learning rates $\gamma$ (with values from 0.01 to 0.4) and $\lambda$ (with values from 0.1 to 1), so we set $\gamma = 0.25$ and $\lambda = 1$ for fast convergence.

The overall optimization process is summarized as follows.

Step 1: Initialize $\omega = 0$ and $b = 0$. Initialize $M$ by extracting the eigenvectors using PCA on the training data. Initialize $g$ as the threshold that makes $s_{ij} > 1$ for most training pairs.

Step 2: Optimize $M$ and $g$ using the stochastic sub-gradient method with Eqs. (12) and (14).

Step 3: Optimize $\omega$ and $b$ using linear SVM with Eqs. (7) and (8).

Steps 2 and 3 are repeated several times to obtain the final values of the model parameters.

B. Creation of Training Data

One important issue that is unexplained in the above learning process is the definition of training pairs $x_i$ and $x_j$ used to optimize Eq. (10). We design two techniques, namely the instance-instance and instance-class constraints, to create the training set.

1) Instance-instance constraint: With this constraint, both $x_i$ and $x_j$ represent the FV descriptors of training images (i.e. instances). Consider that $I$ training samples $\{x_i : i = 1, \ldots, I\}$ are given. The straightforward training approach would be to use all combinations of data pairs as $x_i$ and $x_j$. However, this approach would result in a large amount of $I^2/2$ training pairs. We suggest that a large training set is not suitable for our optimization problem, since with the feature space complexity, a large training set would lead to many contradicting constraints and make it hard for the optimization process to minimize the cost while maintaining its generalizability. Existing distance metric learning methods normally use a small set of training pairs selected randomly or predefined by the problem domain [57], [38]. In our approach, we design a classification score-based pruning technique to select a subset of the training pairs.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.
To do this, a linear SVM classifier is trained on the \( I \) training data using the original high-dimensional FV descriptors. The classification score \( p_{i,l} \) of \( x_i \) for each class \( l \) is then computed as the probability estimate from the SVM classifier. Next, for each class \( l \), three classification thresholds are computed. The first threshold \( th_{1,l} \) is computed by sorting the classification scores \( p_{i,l} \) of all \( x_i \) of class \( l \) and choosing the classification score at the \( c_1 \) percentile:

\[
th_{1,l} = \text{percentile}_{c_1}(p_{i,l} : i = 1, \ldots, I; y_i = l).
\]

The second threshold \( th_{2,l} \) is computed similarly at the \( c_2 \) percentile. The third threshold \( th_{3,l} \) is computed at the \( c_3 \) percentile but for all \( x_i \) not belonging to class \( l \):

\[
th_{3,l} = \text{percentile}_{c_3}(p_{i,l} : i = 1, \ldots, I; y_i \neq l).
\]

These thresholds help to identify \( x_i \) with high classification or mis-classification scores. The \( c_1 \), \( c_2 \), and \( c_3 \) percentile values are constant parameters and the settings are described in Section IV.

Then, three pruning rules are defined based on the classification scores and thresholds to select the training pairs.

**Rule 1:** If a certain \( x_i \) has a classification score \( p_{i,y_i} \) higher than the threshold \( th_{1,y_i} \), this \( x_i \) is not included in the training set. This means that we only include samples with low classification scores into the training set, so that the training will focus on such data.

**Rule 2:** For a selected \( x_i \), if \( x_j \) belongs to the same class as \( x_i \) and \( p_{j,y_j} > th_{2,y_j} \), \( x_j \) is selected to form a training pair with \( x_i \). Here \( p_{j,y_j} > th_{2,y_j} \), means that \( x_j \) is a good positive sample with large classification margin, and we expect \( x_i \) to move closer to \( x_j \) after dimension reduction.

**Rule 3:** For a selected \( x_i \), if \( y_j \neq y_i \) and \( p_{j,y_j} > th_{3,y_j} \), \( x_j \) is selected to form a training pair with \( x_i \). Here \( p_{j,y_j} > th_{3,y_j} \), means that \( x_j \) is a negative sample that could be easily misclassified as class \( y_i \), and \( x_j \) should be better separated from \( x_j \) after dimension reduction.

2) **Instance-class constraint:** This constraint captures the relationship between instances and classes using a different definition of \( x_j \). Our design is inspired by the sparse representation technique. Briefly, with sparse representation classification, a sparse reconstruction of \( x_i \) is obtained using the reference dictionary of each class \( l \), then \( x_i \) is classified to the class corresponding to the smallest reconstruction difference. The difference between \( x_i \) and its sparse reconstruction gives an indication of the similarity between \( x_i \) and the overall class at the global level, and by optimizing this reconstruction difference with distance metric learning, better classification results can be obtained. We thus consider that the sparse reconstruction can be used as \( x_j \), and with the constraint in Eq. (8), we expect to improve the feature space separation between difference classes.

Specifically, for an \( x_j \) that is selected using Rule 1 as described above, \( x_j \) is computed as the sparse reconstruction of \( x_i \) from the training data of class \( y_i \) (excluding \( x_i \)):

\[
x_j = Rv_i
\]

where \( R \in \mathbb{R}^{d \times n_x} \) is the reference dictionary created by concatenating the FV descriptors of all \( n_x \) number of training data of class \( y_i \) (excluding \( x_i \)). The vector \( v_i \in \mathbb{R}^{n_x} \) is the sparse coefficient for reconstructing \( x_i \) as \( x_j \), and is derived using the locality-constrained linear coding (LLC) algorithm [58]:

\[
\argmin_{v_i} \|x_i - Rw_i\|^2 + \beta\|c_i \odot v_i\|^2
\]

\[
s.t. \; \; 1^T v_i = 1, \; \|v_i\|_0 = q
\]

where \( c_i \in \mathbb{R}^{n_x} \) contains the Euclidean distance between \( x_i \) and each descriptor in \( R \), \( \beta = 0.01 \) is a constant scalar, and \( q \) specifies the number of reference descriptors used in the sparse reconstruction. LLC is chosen for its computational efficiency with analytical solution and its general effectiveness in providing a good sparse reconstruction. The resultant \( x_j \) gives a class-level representation of \( x_i \). With small intra-class variation, \( x_j \) would be close to \( x_i \) with small reconstruction difference. Therefore, we can label \( x_j \) as class \( y_j \), and the training pair of \( x_i \) and \( x_j \) encourages the reduction of intra-class variation in the low-dimensional space.

In addition, another \( x_j \) is computed as the sparse reconstruction of \( x_i \) using the training data of the class \( l \neq y_i \). This \( x_j \) represents the resemblance of \( x_i \) with class \( l \), and with \( l \neq y_i \), \( x_i \) is expected to be quite different from \( x_i \). Subsequently, we label this \( x_j \) as class \( l \), and include the training pair of \( x_i \) and \( x_j \) to impose the optimization goal that \( x_i \) should be well separated from \( x_j \) after dimension reduction.

C. **Extension to Multi-class**

For simplicity of presentation, we have described our SDR method as a binary classification model so far. To extend the method to support multi-class classification, several adaptations are required.

First, a one-vs-all SVM classifier is used in place of the binary SVM classification, by learning one set of weight vector \( \omega \) and bias \( b \) for each class. This affects the actual classification with Eq. (4) and the step 3 of the optimization process for parameter learning.

Second, when learning the project matrix \( M \), Eq. (9) is modified to accommodate for multiple classes:

\[
\theta_{ij}(g - \|Mx_i - Mx_j\|^2) + \alpha\{(\omega_{y_i} \cdot (Mx_i) + b_{y_i}) + (\omega_{y_j} \cdot (Mx_j) + b_{y_j})\} \geq 1.
\]

Compared to Eq. (9), here \( \omega \) and \( b \) are class-specific and the class labels \( y_i \) and \( y_j \) become redundant. Step 2 of the optimization process is then updated accordingly.

Third, for the instance-class constraint, when creating \( x_j \) for a selected \( x_i \) from classes \( l \neq y_i \), one sparse reconstruction is computed from each class \( l \neq y_i \). The two reconstructions that are the most similar to \( x_i \) are then used as \( x_j \) to construct two pairs of training data. These \( x_j \) vectors represent the hard negative samples that could potentially cause misclassification due to their high similarities with \( x_i \).

In addition, to speed up the learning process of the model parameters, we first apply PCA to reduce the dimension of FV descriptors (d) to the maximum possible dimension, prior to applying SDR. Our empirical analysis shows that such a preprocessing does not affect the classification performance but largely reduces the training time.
IV. Experimental Setup

For performance evaluation, we used four public datasets. Example images of each dataset are shown in Fig 1.

1) UCSB breast cancer dataset: Breast cancer is one of the most common types of cancer among women [59]. Detection of malignant tumors is the key to treatment and prognosis. Histopathological analysis of tissue morphology, such as the nuclear appearance and pleomorphism, and mitotic activity, typically provides the final diagnosis of benign and malignant tumors. The UCSB breast cancer dataset contains 58 sections from tissue microarray images of 32 benign and 26 malignant breast cancer patients. The images are hematoxylin and eosin (H&E) stained, and each image is of 896 × 768 pixels. This dataset presents a binary classification problem of benign and malignant cases, and is commonly used as the benchmark for multiple instance learning algorithms [13], [60].

2) MICCAI 2015 CBTC challenge dataset: Lower grade gliomas represent a diverse group of primary brain tumors, and astrocytomas and oligodendrogliomas are the most common subtypes of lower grade gliomas [61]. Accurate subtyping of the tumor helps to determine the appropriate treatment options, and patients with proper treatment can have longer-term survival. One task of MICCAI 2015 CBTC challenge is to classify the two subtypes of lower grade glioma from histopathology images. The dataset with released labels contains 32 whole-slide images of varying sizes with 40× apparent magnification. Half of these images are astrocytoma, and the other half are oligodendroglioma. The difficulty in achieving accurate classification automatically is that the morphological features of each tumor subtype are typically heterogeneous although the tumor cells in oligodendroglioma tend to be more uniform and have round nuclei with clear halo.

3) IICBU malignant lymphoma dataset: Lymphoma is a common form of blood cancer, and chronic lymphocytic leukemia (CLL), follicular lymphoma (FL), and mantle cell lymphoma (MCL) are the major types of malignant small B-cell lymphoma [62]. The diagnosis of lymphoma is commonly performed on tissue samples, and the specific type of lymphoma is usually characterized by the cytologic features of the abnormal cells. Distinguishing them is essential in the treatment process, but is also challenging since one lymphoma type can contain varying histological patterns. The IICBU lymphoma dataset contains 374 H&E stained image sections captured using brightfield microscopy, with 113 CLL, 139 FL, and 133 MCL cases. Each image is of 1388 × 1040 pixels. The images are collected from different sites with a large degree of staining variation representing the typical clinical settings. This dataset has been used in various microscopy image classification studies [7], [63], [11], [64].

4) IICBU RNAi dataset: RNA interference (RNAi) is a biological process that RNA molecules inhibit gene expression or translation, and facilitates the systematic disruption of gene expression for the study of gene functions in the cell [65]. The IICBU RNAi dataset contains 200 fluorescence microscopy images of fly cells, acquired using a light microscope with 60× objective. The cells are stained with DAPI for visualizing the nuclei. The images show phenotypes resulting from knockdown of a particular gene using RNAi. Ten genes each with 20 images are selected to form the dataset. The gene IDs are CG1258, CG3733, CG3938, CG7922, CG8114, CG10873, CG12284, CG17161, CG8222, and CG9484. Each image is of 1024 × 1024 pixels. Classification of the gene classes helps to quantify the phenotype differences between the different genes, and automated classification provides a means to support the high-throughput screening analysis [66], [67].

For all datasets, four-fold cross validation was performed. Within each round, three quarters of images were used for training and the remaining one quarter for testing. This protocol follows the standard setting used in UCSB breast cancer studies [13], [60] and is similar to the benchmark on IICBU RNAi dataset [67]. Note that each image in the UCSB and CBTC datasets comes from a different patient, hence it is guaranteed that the training and testing sets contain images from different patients. It is however not clear if the images in the lymphoma and RNAi datasets are all from different subjects. Nevertheless, cross validation has commonly been performed for these two datasets [11], [66], [67]. Classification accuracy and the receiver operating characteristic (ROC) analysis were used as the performance metrics.

Most of the parameters were set in the same way for all datasets. Specifically, the dimension $h$ of the dimension reduced descriptor was set to half of the number of images in the specific dataset. For example, the UCSB breast cancer dataset contained 58 images, and the original SIFT, DBN, and CNN based FV descriptors were of 8192, 8192, and 65536 dimensions, respectively. With SDR, the reduced feature dimension $h$ became 29 for all three feature types. The parameter $\alpha$ in Eq. (9) is set to 0.1, so that a higher weight is allocated to the distance factor. The number of iterations for optimizing the model parameters was three. The sparsity factor $q$ in Eq. (18) was set to 3 and 5, so that two different reconstructions were obtained and the number of instance-class based training samples was doubled. The $c_3$ parameter used to compute the score threshold was set to 0.95, in order to reduce the number of negative samples. The only dataset-specific parameters were $c_1$ and $c_2$, which were also set differently for different local features (SIFT, DBN, and CNN), with values of 0.4, 0.6 or 0.8. These settings were determined based on ten-fold cross validation within the training set.

Our program was developed in Matlab R2015b, running on a PC with an Intel i7 CPU and 16GB memory. The DeeBNet [47], MatConvNet [68], and VFLFeat [69] packages were used to compute the DBN and CNN features, and FV encoding. In addition, due to the large size of the whole-slide images in the CBTC dataset, the images were resized to 1/4 of the original size to reduce the computational cost. Also, when extracting the local CNN features, an image was subdivided into 16 sections and the CNN model was applied to each section separately. The extracted local features were then combined during FV encoding.

V. RESULTS AND DISCUSSION

A. Overall Performance

Our classification results are shown in Table I. The results were obtained using FV descriptors of different local features:
dense SIFT, patch-wise DBN, and CNN features from the last convolutional layer, which are named as FV-SIFT, FV-DBN, and FV-CNN. The three FV descriptors were then combined into longer descriptors: S+D (combining FV-SIFT and FV-DBN), S+C (combining FV-SIFT and FV-CNN), D+C (combining FV-DBN and FV-CNN), and S+D+C (combining all three FV descriptors). Classification using linear kernel SVM without dimensionality reduction, and classification by first applying our SDR method, were evaluated. The classification performance was measured using the overall accuracy and area under the curve (AUC). For the multi-class problem in the IICBU lymphoma and RNAi datasets, ROC analysis was conducted in a one-vs-all manner, and the AUCs from all binary classifications were averaged to produce the overall AUC. The three best results from existing studies are included for comparison as well.

The results demonstrate the benefit of our SDR method. The consistent improvement with SDR on all datasets indicates the advantage of reducing the feature dimensionality especially for datasets of relatively small sizes. The largest performance improvement with SDR was obtained on the CBTC dataset. For example, with SVM classification of the original high-dimensional FV-CNN descriptors, only 34.4% accuracy was obtained; however, after dimension reduction with SDR, the accuracy improved to 71.9%. This suggests that there is high redundancy in the FV-CNN descriptors for the CBTC dataset. With the small amount of images available, dimension reduction helps to enhance the discriminative power of descriptors while avoiding over-fitting of the trained classifier. Similarly large improvement was obtained with the other FV descriptors and combination of descriptors as well. On the RNAi dataset, however, the degree of improvement provided by SDR was relatively small with around 1% increase in accuracy. We suggest that this was mainly due to the large number of classes (i.e. 10) in this dataset. In general, multi-class classification is more difficult than binary classification, since it is more complex to establish a clear separation in the feature space between a larger number of classes. With our simple extension to multi-class problem, the optimization goal imposes a stringent requirement on the feature space separation, which is hard to achieve with the large number of conflicting constraints from multiple classes. A one-vs-all type of model learning, similar to the technique used in multi-class SVM, could potentially improve the results further. However, this type of design would be more complex involving fusion and calibration, hence in our current study we focused on the primal definition of the SDR model and kept the multi-class handling simple.

We obtained better classification results over the state-of-the-art on these datasets. The existing studies on the UCSB dataset used AUC as the performance metric, while those on the lymphoma and RNAi datasets used the accuracy measure. To the best of our knowledge, there are no published results for the CBTC dataset. Our method obtained near perfect (0.999) AUC on the UCSB dataset when the FV-CNN descriptor was used, while the state-of-the-art [70] (based on joint clustering and classification in a multiple instance learning framework) was 0.95. For the lymphoma dataset, the current state-of-the-art [64] proposed to use a combination of LBP and CNN features based on segmented regions with an ensemble of SVM classifiers, and provided 95.5% accuracy, while we achieved 97.9% accuracy with the S+C descriptor. For the RNAi dataset, we obtained 96% accuracy with the S+D+C descriptor, while the state-of-the-art [67] was 92% based on customized feature representations and ensemble classification.

We suggest that the low classification accuracy on the CBTC dataset are mainly related to the complexity of information extraction from whole-slide images and the difficulty with

### Table I

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>Metric</th>
<th>Single FV descriptor</th>
<th>Combination of FV descriptors</th>
<th>Top-3 existing studies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>FV-SIFT</td>
<td>FV-DBN</td>
<td>FV-CNN</td>
</tr>
<tr>
<td>UCSB</td>
<td>SVM</td>
<td>Accuracy</td>
<td>87.9</td>
<td>82.8</td>
<td>96.6</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>Accuracy</td>
<td>89.7</td>
<td>89.7</td>
<td>98.3</td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>AUC</td>
<td>0.906</td>
<td>0.915</td>
<td>0.986</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>AUC</td>
<td>0.925</td>
<td>0.928</td>
<td>0.999</td>
</tr>
<tr>
<td>CBTC</td>
<td>SVM</td>
<td>Accuracy</td>
<td>53.1</td>
<td>62.5</td>
<td>34.4</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>Accuracy</td>
<td>59.4</td>
<td>75.0</td>
<td>71.9</td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>AUC</td>
<td>0.500</td>
<td>0.551</td>
<td>0.262</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>AUC</td>
<td>0.535</td>
<td>0.723</td>
<td>0.711</td>
</tr>
<tr>
<td>Lymphoma</td>
<td>SVM</td>
<td>Accuracy</td>
<td>90.4</td>
<td>88.8</td>
<td>93.9</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>Accuracy</td>
<td>93.3</td>
<td>90.1</td>
<td>95.7</td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>AUC</td>
<td>0.961</td>
<td>0.951</td>
<td>0.984</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>AUC</td>
<td>0.970</td>
<td>0.960</td>
<td>0.991</td>
</tr>
<tr>
<td>RNAi</td>
<td>SVM</td>
<td>Accuracy</td>
<td>86.5</td>
<td>80.5</td>
<td>88.5</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>Accuracy</td>
<td>88.0</td>
<td>81.5</td>
<td>90.5</td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>AUC</td>
<td>0.975</td>
<td>0.957</td>
<td>0.983</td>
</tr>
<tr>
<td></td>
<td>SDR+SVM</td>
<td>AUC</td>
<td>0.982</td>
<td>0.961</td>
<td>0.989</td>
</tr>
</tbody>
</table>
morphological distinction of lower grade gliomas. In particular, not all areas of a whole-slide image would be pathological relevant. Therefore, encoding the global image-level information in the FV descriptor would include noise that could affect the discriminative power of the feature descriptor. This could also explain why by reducing the feature redundancy with SDR, the classification performance on CBTC improves greatly over using the original FV descriptor. Also, from the morphological perspective, astrocytomas encompass a range of histopathological patterns with a number of subtypes such as the diffuse astrocytoma and anaplastic astrocytoma, and this heterogeneity could make it particularly difficult to establish a compact feature space for astrocytomas. Subsequently, this would affect the classification between astrocytoma and oligodendroglioma as well.

Different from the CBTC dataset, images in the UCSB and lymphoma datasets represent manually selected tissue sections of the tumor samples. This selection process could have helped to identify the pathological relevant areas, and our feature representation could thus summarize the important histopathological patterns more effectively. The high classification accuracies on the UCSB and lymphoma datasets also imply that the morphological differences between benign and malignant breast tumors and between the three malignant lymphoma types are more pronounced, particularly on the range of cases present in the datasets. In addition, note that while cell-level features are important to identify malignant breast tumor or the specific malignant lymphoma type, our FV descriptor represents the visual characteristics of the entire image. This is different from the existing approaches [13], [60], [70], [64], which involve cell segmentation and cell-based feature extraction. Without this cell segmentation process, our method can thus be more easily applied to different types of images and applications.

The RNAi dataset presents a very different application domain, with images showing phenotypes after the knock-down of a particular gene. Our results demonstrate that although the images contain lots of background, the FV descriptor could well represent the overall phenotype. In addition, we would like to note that although the images in this dataset are acquired using fluorescence microscopy, the classification problem on this dataset is not necessarily easier than the other histopathological images. For example, the classification performance on the RNAi dataset was much lower than that on the lymphoma dataset, when the same approach was used [7], [11]. We suggest that this is mainly due to the complexity of the 10-class classification problem, and the intrinsic visual similarity between different phenotypes. Nevertheless, we achieved good improvement over the state-of-the-art [67], indicating the generalizability and effectiveness of our method when applied to a non-histopathological problem.

The results in Table I also show that the choice of descriptor has a high impact on the classification performance, and the effect of descriptors varies on different datasets. For example, FV-CNN provided better results than FV-SIFT and FV-DBN on the UCSB, lymphoma, and RNAi datasets, but the worst for the CBTC dataset. Also, FV-DBN was the most effective descriptor on the CBTC dataset, but the least effective for the other datasets. Since the FV-DBN descriptor involves dataset-specific learning of the DBN model, such results imply that there is some special characteristic in the CBTC images that can be better discovered with customized learning while the other datasets benefit more from the pretrained CNN models. In addition, with the combination of FV descriptors, the performance improvement on the lymphoma and RNAi datasets was prominent, indicating the advantage of incorporating multimodal descriptors of complementary information to represent the images.

Overall, we suggest that using S+D+C can provide high classification performance for all datasets consistently. Fig. 3 shows an example image from each dataset that was misclassified when S+D+C was used as the descriptor. For example, the image shown from the UCSB dataset is one of the benign cases misclassified as malignant. Compared to the correctly classified benign cases, this image exhibits a higher cellularity with a large amount of cells, which would often indicate malignancy. The image from the CBTC dataset is an astrocytoma image that was misclassified as oligodendrogliomas. A closer look at the high-resolution image shows that there are many halos in the image, which is a typical factor confusing the identification of astrocytomas. The image from the lymphoma dataset shows an example CLL image that was misclassified as MCL. As we mentioned, the lymphoma images were collected from different sites with large variations; and we noticed that this particular image had quite low resolution. Consequently, the nuclear features were hard to visualize and the overall image shows a similar pattern to the MCL images. This could thus contribute to the misclassification. The image shown from the RNAi dataset is the only image in class CG1258 that was misclassified, and it was labeled as class CG3733. This could
Fig. 4. Classification accuracy comparing FV with BOW encoding, when different local features are used and SVM classification without dimensionality reduction is performed. The feature vector from the penultimate fully-connected layer of the CNN model is also included for comparison (C-FC).

Fig. 5. The average change in classification accuracy after applying dimensionality reduction with different FV descriptors, compared to SVM classification without dimensionality reduction.

imply some similarities between the two phenotypes.

The computational cost of our method was evaluated by the training and testing phases. At the training phase, the process flow includes local feature extraction as training data for GMM, generation of GMM, FV encoding of the local features with the learned GMM, learning of projection matrix, dimensionality reduction of the FV descriptor with the learned projection matrix, and learning of linear-kernel SVM classifier. The most time-consuming part is the local feature extraction. For example, with the CNN approach, about 2.5 minutes per image were required to extract the local features on the CBTC dataset, and on average 9.1 seconds per image on the other three datasets; and GMM generation took about 7 seconds on each dataset. The time required for learning the projection matrix was roughly proportional to the number of training data, with about 2.3 seconds, 1.9 seconds, 18.5 seconds, and 8.1 seconds on the UCSB, CBTC, lymphoma, and RNAi datasets, respectively. SIFT and DBN local feature extractions were faster than the CNN approach, requiring about 22 and 78 seconds per image on the CBTC dataset, and 1.3 and 4.5 seconds on the other three datasets. With the smaller dimensionality of SIFT- and DBN-based FV descriptors, the time required for generating the projection matrix was reduced as well. The other processes including the FV encoding, dimensionality reduction, and learning of SVM were fast and required negligible time. During the testing phase, the local feature extraction was still the most time-consuming process, and the time required for FV encoding, dimensionality reduction, and SVM classification was negligible.

B. Evaluation of FV Descriptor

The effects of FV descriptors in comparison with other more standard feature representations are shown in Fig. 4. The results were obtained using SVM classification without dimensionality reduction. For comparison, the various local features (SIFT, DBN, and CNN) were integrated with bag-of-words (BOW) encoding to evaluate the benefit of FV encoding. We chose to compare with BOW since it represented a conventional feature encoding technique. To compute the BOW descriptors, we experimented with $64, 128, \ldots, 2048$ feature words and found 1024 to provide the best overall results. The 4096-dimensional CNN descriptor derived from the penultimate fully-connected layer (namely C-FC) was included as well. This C-FC shows the typical usage of CNN in classification when pretrained models are used [42].

The findings can be summarized in three points. (i) FV encoding is more effective than BOW encoding for the UCSB, lymphoma, and RNAi datasets, but not for the CBTC dataset. The disadvantage of FV encoding is especially evident with the CNN local features on the CBTC dataset. This indicates that on this small dataset, the high dimension of FV-CNN descriptor gave limited discriminative and representative capability, while the BOW encoding of CNN local features
C. Evaluation of Dimensionality Reduction

Fig. 5 shows the performance comparison between our SDR method and the other dimensionality reduction techniques. We compared with the popular PCA, LDA, and ISOMAP techniques. GDA, as an improved version of LDA, was also included. FML is similar to SDR that the projection matrix is obtained based on distance learning. However, different from SDR, FML does not integrate the distance learning with the SVM optimization. Also, while we define the training pairs with instance-instance and instance-class constraints, FML is originally designed for face recognition and uses training pairs that are selected manually. To apply FML to the microscopy image datasets, we included all training pairs without pruning. For PCA, ISOMAP, and FML, we reduced the feature dimension to half of the number of images, the same as our SDR method. For LDA and GDA, the feature dimension was reduced to the number of image classes. Linear SVM classification was used with all compared approaches. Note that these approaches were applied to each FV descriptor, i.e., FV-SIFT, FV-DBN, FV-CNN, S+D, S+C, D+C, and S+D+C. For easier visualization of the results, the difference between the classification accuracy obtained using the compared approach (e.g., PCA then SVM) and the accuracy obtained using SVM without dimensionality reduction was computed, and the differences from all descriptors were averaged as the performance indicator.

The results show that our SDR method outperformed the compared approaches on all datasets. LDA was the second best, providing performance improvement on three datasets except the RNAi dataset. While GDA is a kernelized version of LDA, it actually led to lower performance than LDA except for the lymphoma dataset. This implies that when the size of dataset is small, the linear model in LDA would be more suitable than the kernelized model in GDA. In general, LDA and GDA performed better than PCA and ISOMAP. This indicates the advantage of having supervised dimensionality reduction for our classification problems. However, although FML is also a supervised algorithm, it resulted in the largest performance degradation on the UCSB, lymphoma, and RNAi datasets, and the smallest improvement on the CBTC dataset. We suggest that this was mainly due to the large number of training pairs imposing many contradicting constraints and affecting the optimization outcome.

To further evaluate our SDR method, we analyzed the effects of the various components in SDR. In particular, we tested the following approaches: (i) II-only: only the instance-instance constraint was implemented to construct the training pairs and the instance-class constraint was omitted; (ii) All-P: instead of the instance-instance constraints, all pairs of training data were included; (iii) IC-only: only the instance-class constraint was implemented to construct the training pairs and the instance-instance constraint was omitted; (iv) Iter-1: only one iteration was performed to optimize the model parameters, which was equivalent to learning the dimension reduction model based on the distance metric only without integrating the discriminative objective of SVM. These experiments were conducted using the FV-SIFT, FV-DBN, and FV-CNN descriptors.

The results are shown in Fig. 6. The advantage of SDR over II-only and IC-only demonstrates the usefulness of having both the instance-instance and instance-class constraints for creating the training data. All-P provided lower performance than II-only in most cases, indicating the advantage of the classification score-based pruning technique. With Iter-1, the classification accuracy sometimes improved over using SVM without dimensionality reduction and sometimes became lower,
and with more iterations that integrated the distance metric learning with the discrimination objective of SVM, the classification performance gradually increased and became stable after three iterations. Overall, these results indicate the effect of the various components of our SDR method, and the need of combining them into one formulation.

The most important parameters in our SDR method are the percentiles $c_1$, $c_2$, and $c_3$ used to calculate the thresholds for training data selection. We conducted cross-validation within the training set and chose the parameters providing the best overall performance, and the settings can vary depending on the dataset and type of local feature. For example, with FV-CNN, $c_1$ was set to 0.8, 0.4, 0.6, and 0.4 for the UCSB, CBTC, lymphoma, and RNAi datasets datasets, respectively. When $c_1$ was set to 0.4, 0.6, 0.4, and 0.8, the lowest classification performance was obtained with 1.7%, 9.4%, 1.1%, and 2.5% drop in classification accuracy on the four datasets, respectively. The results show that the parameter setting indeed has an important effect on the classification performance. This is because these parameters control the feature space of training data, which in turn determines the optimization constraints imposed for learning the projection matrix.

VI. CONCLUSIONS AND FUTURE STUDY

We present a microscopy image classification method in this paper. The images are represented by FV descriptors generated based on dense SIFT, patch-based DBN that are learned in an unsupervised manner, and CNN local features that are extracted from the last convolutional layer using the pretrained VGG-VD model. To further enhance the discriminative power of the FV descriptors and decrease the impact of small data size on the classification performance, a separation-guided dimension reduction (SDR) method is designed to reduce the dimension of the FV descriptors while improving the feature space separation between classes. The resultant low dimensional descriptors are finally classified using a linear-kernel SVM to label the images. We experimented on four publicly available microscope image dataset and obtained performance improvement over the state-of-the-art. We demonstrate the advantage of our SDR method over the commonly used dimensionality reduction techniques, and also the benefit of FV encoding for representing microscopy images.

As a future study, we will investigate designing a customized patch-based CNN model to be used as the local feature extractor. We will also investigate improving the multi-class handling with SDR, such as using a one-vs-all model to learn multiple projection matrices and fusing the binary classification outputs to obtain the final results. Furthermore, another potential application area of our method is large scale biomedical image retrieval. When retrieving images from a large and heterogeneous cohort, a feature representation technique that can be generally effective with minimum domain-specific processing would be very useful. The training of SDR can also be naturally tuned to fit the retrieval objective, since the underlying formulation is based on distance metric learning. It will be interesting to see how our method works in such a domain.

REFERENCES


