

Cell Encoding for Histopathology Image Classification

Xiaoshuang Shi¹, Fuyong Xing^{1,2}, Yuanpu Xie¹, Hai Su¹, and Lin Yang^{1,2}(✉)

¹ J. Crayton Pruitt Family Department of Biomedical Engineering,
University of Florida, Gainesville, FL 32611, USA
lin.yang@bme.ufl.edu

² Department of Electrical and Computer Engineering, University of Florida,
Gainesville, FL 32611, USA

Abstract. Although many image analysis algorithms can achieve good performance with sufficient number of labeled images, manually labeling images by pathologists is time consuming and expensive. Meanwhile, with the development of cell detection and segmentation techniques, it is possible to classify pathology images by using cell-level information, which is crucial to grade different diseases; however, it is still very challenging to efficiently conduct cell analysis on large-scale image databases since one image often contains a large number of cells. To address these issues, in this paper, we present a novel cell-based framework that requires only a few labeled images to classify large-scale pathology ones. Specifically, we encode each cell into a set of binary codes to generate image representation using a semi-supervised hashing model, which can take advantage of both labeled and unlabeled cells. Thereafter, we map all the binary codes in one whole image into a single histogram vector and then learn a support vector machine for image classification. The proposed framework is validated on one large-scale lung cancer image dataset with two types of diseases, and it can achieve 87.88% classification accuracy on 800 test images using only 5 labeled images of each disease.

1 Introduction

Effective and efficient image analysis plays an important role in computer-aided diagnosis (CAD) in digital pathology. Traditional image analysis algorithms [10, 17] and deep learning based methods [4, 11] have achieved satisfactory performance in many applications, while they usually require numerous labeled pathology images to learn a robust and stable model. Manually labeling images by doctors or pathologists is labor intensive, time-consuming and even error-prone. In addition, the differentiation of many diseases grades relies on the cellular information such as shape, area, nuclear and cytoplasm appearances [18, 20]. Thus individual cell analysis might be significantly beneficial to analyzing pathology images. Although the development of cell detection and segmentation techniques enables the cell information extraction, it is still a very challenging task because

(i) one single digitized specimen usually contains hundreds or thousands of cells; (ii) segmented cells usually exhibit low inter-class but high intra-class variations and contain considerable image noise. To address these issues, in this paper, we focus on using a hashing tool to encode each cell into binary codes for pathology image classification.

Hashing is to encode the high-dimensional data into compact binary codes with preserving the similarity among neighbors. It is usually used to retrieve nearest neighbors based on a certain similarity measurement in large-scale databases [7, 15], because of the gain in computation and storage. Over the past few years, many hashing algorithms [9, 12–14] have been proposed or developed in the literature. Generally, these algorithms can be classified into two major groups: (i) Unsupervised hashing that explores intrinsic low-dimensional structures of the data to preserve the similarity without any supervision; (ii) Supervised hashing that uses semantic information to assist retrieval and searching. Due to the semantic gap, supervised hashing algorithms are usually employed to classify pathology images. Some supervised hashing algorithms [6, 7, 19, 20] have shown superior classification performance to traditional methods, however, they usually have one or two following major limitations: (i) Directly encoding the whole pathology image rather than individual cells into binary codes for image classification might fail to effectively encode the cellular features; (ii) A large number of labeled data is required to achieve promising accuracy.

Motivated by the aforementioned observations, in this paper, we propose a novel cell-based framework that makes use of only few labeled images for disease classification. Specifically, we propose a semi-supervised hashing model to encode each cell into a set of binary codes. The proposed model can utilize the semantic information and meanwhile explore the intrinsic low-dimensional structures of cells. Next, we map the binary codes of cells in one whole image into a single histogram vector, upon which we learn a popular classifier, support vector machine (SVM), for image (disease) classification. Fig. 1 shows the main

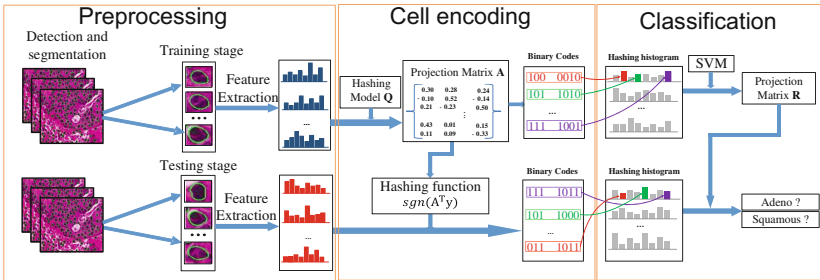


Fig. 1. The flowchart of the proposed framework for pathology image classification. (In the hashing histogram, each vector is a single histogram generated by binary codes of all cells in one whole image, one cell corresponds to one position that is obtained based on binary codes, and the number of cells with the same binary codes corresponds to the bin in one histogram vector.)

idea of the proposed framework. Experiments on thousands lung cancer images demonstrate the effectiveness and efficiency of the proposed framework.

2 Methodology

2.1 Semi-supervised Kernel Discrete Hashing

Definitions and Notations: Given data $\mathbf{X} \in \mathbb{R}^{n \times d}$ that includes n_1 labeled $\mathbf{X}_l \times \mathbb{R}^{n_1 \times d}$ and n_2 unlabeled data points $\mathbf{X}_u \in \mathbb{R}^{n_2 \times d}$, where $n_1 + n_2 = n$. Let $\phi : \mathbb{R}^d \leftarrow \mathcal{H}$ be a kernel mapping from the original space to the kernel space, where \mathcal{H} is a Reproducing Kernel Hilbert Space (RKHS) with a kernel function $\kappa(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y})$. Selecting m ($m \ll n_1$) data points selected from the labeled data (or using K-means to generate m points) as anchors and giving a labeled training data point \mathbf{x} , same as [7, 13], its k -th ($1 \leq k \leq r$) hashing function is defined as:

$$h_k(\mathbf{x}) = \text{sgn}\left(\sum_{j=1}^m \kappa_l(\mathbf{x}_j, \mathbf{x}) a_{kj} - b_k\right) = \text{sgn}(\mathbf{a}_k \bar{\mathbf{K}}_l(\mathbf{x})), \quad (1)$$

where $\mathbf{x} \in \mathbf{X}$, \mathbf{a}_k is the k -th row vector of a projection matrix $\mathbf{A} \in \mathbb{R}^{r \times m}$ and $b_k = \frac{1}{n_1} \sum_{i=1}^{n_1} \sum_{j=1}^m \kappa_l(\mathbf{x}_j, \mathbf{x}_i) a_{kj}$, which suggests $\sum_{i=1}^{n_1} \bar{\kappa}_l(\mathbf{x}_i) = 0$. $\bar{\kappa}_l(\mathbf{x}_i) \subset \bar{\mathbf{K}}_l \in \mathbb{R}^{m \times n_1}$ that is constructed by \mathbf{X}_l and m anchors. Same as [7], we define $h_k(x_i) = h_k(x_j)$ if $(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{M}$ and $h_k(x_i) \neq h_k(x_j)$ if $(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{C}$, where \mathcal{M} is a set that contains the pairs sharing the same class label, and the set \mathcal{C} contains the pairs with different class labels. Let the r -bit hash codes of \mathbf{x} be $\text{code}_r(\mathbf{x}) = [h_1, h_2, \dots, h_r]$. Then if $(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{M}$, $\text{code}_r(\mathbf{x}_i) \circ \text{code}_r(\mathbf{x}_j) = r$; if $(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{C}$, $\text{code}_r(\mathbf{x}_i) \circ \text{code}_r(\mathbf{x}_j) = -r$. The pairwise label matrix $\mathbf{S} \in \mathbb{R}^{n_1 \times n_1}$ is defined as:

$$s_{ij} = \begin{cases} 1 & (x_i, x_j) \in \mathcal{M} \\ -1 & (x_i, x_j) \in \mathcal{C} \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Formulation: Based on the pairwise label matrix in Eq. (2), we attempt to learn a projection matrix \mathbf{A} such that $\text{code}_r(\mathbf{x}_i) \circ \text{code}_r(\mathbf{x}_j) \in [-r, r]$, where \mathbf{x}_i and \mathbf{x}_j are labeled data. In addition, we want to enforce the r -bit hashing code to be mutually uncorrelated so that the redundancy among different bits is minimized, and maximize the information of each bit by $\sum_{i=1}^{n_1} h_k(\mathbf{x}_i) = 0$ ($1 \leq k \leq r$). Moreover, we also want to utilize the intrinsic structure information of all training data including the labeled and unlabeled kernelized data. Therefore, we propose an optimization model as follows:

$$\begin{aligned} \min_{\mathbf{A}} \quad & \|\mathbf{H}^T \mathbf{H} - \mathbf{S}\|_F^2 - \eta \text{Tr} \{ \mathbf{A} \bar{\mathbf{K}} \bar{\mathbf{K}}^T \mathbf{A}^T \}, \\ \text{s.t.} \quad & \mathbf{H} \mathbf{H}^T = n_1 \mathbf{I}_r, \mathbf{H} \mathbf{1}_{n_1} = 0, \end{aligned} \quad (3)$$

where $\mathbf{H} = \text{sgn}(\mathbf{A} \bar{\mathbf{K}}_l)$ and $\mathbf{1}_{n_1} \times \mathbb{R}^{n_1}$ is a column vector with all elements being one. The constraint is only on labeled data because they are more important than

unlabeled data. Since the problem in Eq. (3) is highly non-convex, it is difficult to be directly solved. To learn the projection matrix \mathbf{A} , Eq. (3) is usually relaxed by using a symmetric relaxation strategy [13, 15]. Then it becomes:

$$\begin{aligned} \max_{\mathbf{A}} \text{Tr} \{ \mathbf{A} \bar{\mathbf{K}}_l \mathbf{S} \bar{\mathbf{K}}_l^T \mathbf{A}^T + \eta \mathbf{A} \bar{\mathbf{K}} \bar{\mathbf{K}}^T \mathbf{A}^T \}, \\ \text{s.t. } \mathbf{A} \bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T \mathbf{A}^T = n_1 \mathbf{I}_r, \end{aligned} \quad (4)$$

Let $\mathbf{W} = \bar{\mathbf{K}}_l \mathbf{S} \bar{\mathbf{K}}_l^T + \eta \bar{\mathbf{K}} \bar{\mathbf{K}}^T$, and then Eq. (4) is equivalent to:

$$\begin{aligned} \max_{\mathbf{A}} \text{Tr} \{ \mathbf{A} \mathbf{W} \mathbf{A}^T \}, \\ \text{s.t. } \mathbf{A} \bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T \mathbf{A}^T = n_1 \mathbf{I}_r, \end{aligned} \quad (5)$$

Let $\mathbf{A} \bar{\mathbf{K}}_l = \mathbf{Y}$, and then it is easy to obtain $\mathbf{A} = \mathbf{Y} \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1}$. In practice we obtain \mathbf{A} by $\mathbf{A} = \mathbf{Y} \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T + \epsilon \mathbf{I}_m)^{-1}$ in order to attain stable solutions. Substituting it into Eq. (5), we have:

$$\begin{aligned} \max_{\mathbf{Y}} \text{Tr} \{ \mathbf{Y} \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \mathbf{W} (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \bar{\mathbf{K}}_l \mathbf{Y}^T \}, \\ \text{s.t. } \mathbf{Y} \mathbf{Y}^T = n_1 \mathbf{I}_r, \mathbf{Y} \mathbf{1}_{n_1} = 0. \end{aligned} \quad (6)$$

If we solve Eq. (6) by using ‘relaxation+ rounding’ schemes employed in [15] [7], the discrete matrix \mathbf{H} and its relaxed continuous matrix \mathbf{Y} will produce large accumulated quantization errors, thereby decreasing the quality of binary codes. To reduce the errors, similar to [13], we preserve the discrete constraint $\mathbf{H} = \text{sgn}(\mathbf{Y})$ in the objective, and reformulate Eq. (6) as:

$$\begin{aligned} \max_{\mathbf{Y}, \mathbf{H}} \text{Tr} \{ \mathbf{H} \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \mathbf{W} (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \bar{\mathbf{K}}_l \mathbf{Y}^T \}, \\ \text{s.t. } \mathbf{Y} \mathbf{Y}^T = n_1 \mathbf{I}_r, \mathbf{Y} \mathbf{1}_{n_1} = 0, \mathbf{H} = \text{sign}(\mathbf{Y}), \end{aligned} \quad (7)$$

where $\mathbf{1}_{n_1} \in \mathbb{R}^{n_1}$ is a column vector with all elements being one. Equation (7) is our proposed semi-supervised kernel discrete hashing model.

Let $\mathbf{M} = \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \mathbf{W} (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \bar{\mathbf{K}}_l$, and then Eq. (7) can be rewritten as:

$$\begin{aligned} \max_{\mathbf{Y}, \mathbf{H}} \text{Tr} \{ \mathbf{H} \mathbf{M} \mathbf{Y}^T \}, \\ \text{s.t. } \mathbf{Y} \mathbf{Y}^T = n_1 \mathbf{I}_r, \mathbf{Y} \mathbf{1}_{n_1} = 0, \mathbf{H} = \text{sign}(\mathbf{Y}). \end{aligned} \quad (8)$$

Since Eq. (8) is similar to the objective function of KSDH_H in [13], it can be easily solved. To be clear, we present the optimization procedure of Eq. (8) in Algorithm 1, which is named as semi-supervised kernel discrete hashing (SSKDH). It is worth noting that when $\eta = 0$, SSKDH will degenerate to KSDH_H.

2.2 Hash Codes Based Classification

After learning the projection matrix \mathbf{A} and encoding each cell into r -bit binary codes, intuitively, we can classify each cell and then use major voting to attain the label of each image, however, empirically this strategy is usually ineffective for encoded cells. Because the small number of training cells leads to bad cell

Algorithm 1. SSKDH

Input: Kernelized data matrix $\bar{\mathbf{K}} \in \mathbb{R}^{m \times n}$ including $\bar{\mathbf{K}}_l$ and $\bar{\mathbf{K}}_u$, weight matrix \mathbf{S} , number of bits r , and parameters η and ϵ

Output: Projection matrix $\mathbf{A} \in \mathbb{R}^{r \times m}$

Initialize: $t=0$, let \mathbf{A}_0 be the eigenvectors of $\bar{\mathbf{K}}_l \mathbf{S} \bar{\mathbf{K}}_l^T$ and $\mathbf{H}_0 = \text{sgn}(\mathbf{A}_0 \bar{\mathbf{K}}_l)$;

Calculate $\mathbf{W} = \bar{\mathbf{K}}_l \mathbf{S} \bar{\mathbf{K}}_l^T + \eta \bar{\mathbf{K}} \bar{\mathbf{K}}^T$, $\mathbf{M} = \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \mathbf{W} (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T)^{-1} \bar{\mathbf{K}}_l$,

$\mathbf{J} = \mathbf{I}_{n_1} - \frac{1}{n_1} \mathbf{1}_{n_1} \mathbf{1}_{n_1}^T$, and $\mathbf{P} = \mathbf{J} \mathbf{M}$;

while not converge or reach the maximum iterations

 Calculate \mathbf{U}_{t+1} , $\bar{\mathbf{U}}_{t+1}$ and \mathbf{V}_{t+1} based on Proposition 1 in [13];

 Update $\mathbf{Y}_{t+1} = \sqrt{n} \mathbf{V}_{t+1} [\mathbf{U}_{t+1}, \bar{\mathbf{U}}_{t+1}]^T$;

 Update $\mathbf{H}_{t+1} = \text{sgn}(\mathbf{Y}_{t+1})$;

end while

Calculate $\mathbf{A} = \mathbf{Y} \bar{\mathbf{K}}_l^T (\bar{\mathbf{K}}_l \bar{\mathbf{K}}_l^T + \epsilon \mathbf{I}_m)^{-1}$

classification accuracy (For the binary classification problem, the accuracy of one class is usually larger than 50%, while the accuracy of the other class is smaller than 50%). In order to obtain robust and stable classification accuracy, we map all cell information of one image into a 2^r -dimensional histogram vector $\mathbf{v}^I \in \mathbb{R}^{2^r}$, which includes cell information to represent the whole image. The r -bit binary codes of one cell can be transformed into an integer corresponding to one index in the 2^r -dimensional vector, and the coefficient in the index is the number of cells with the same integer. For example, given one cell x with r -bit binary codes $[h_1, h_2, h_3, \dots, h_r]$, it corresponds to the index:

$$idx = 1 + \sum_{k=1}^r h_k 2^{k-1}. \quad (9)$$

Suppose that p cells corresponds to the index idx in one image, then $\mathbf{v}^I(idx) = p$.

After transforming all labeled images into 2^r -dimensional histogram vectors, we normalize each vector to be unit and then employ an SVM to learn model parameters. For clarity, we present the detailed procedure in Algorithm 2, which is named as hashing codes for classification (HCC) in this paper. Note that HCC is different from [11, 20], because HCC essentially adopts SVM to weight each cell in one whole image.

3 Experiments

In order to evaluate the proposed framework, we conduct extensive experiments on one lung cancer image dataset with two types of diseases: adenocarcinoma and squamous cell carcinoma. We collect the lung cancer images from The Cancer Genome Atlas (TCGA) [5], which consists of 610 adenocarcinoma and 630 squamous images. They contain about 589K adenocarcinoma and 1248K squamous cells, respectively. We detect and segment these cells by using the method in [16]. And then crop all cells as image patches with the corresponding labels. Next, we extract 1024 features from each cell by using HOG and GIST descriptors, respectively.

Algorithm 2. HCC

Input: Binary codes of labeled images $\{\mathbf{H}_1; \mathbf{H}_2; \mathbf{H}_3; \dots; \mathbf{H}_{N_1}\}$ and the corresponding labels $\mathbf{L} \times \mathbb{R}^{N_1}$, where $\mathbf{H}_i \in \mathbb{R}^{q_i \times r}$ ($1 \leq i \leq N_1$) represents binary codes of q_i cells in the i -th image, and the number of bits r

Output: Projection matrix \mathbf{R}

Initialize: $\mathbf{V}^I = \mathbf{0}$, where $\mathbf{V}^I \in \mathbb{R}^{N_1 \times 2^r}$ is the mapped matrix of labeled images;

for $i = 1$ to N_1 do

 for $j = 1$ to q_i do

$idx \leftarrow 1 + \sum_{k=1}^r \mathbf{H}_{ij} k 2^{k-1}$,

$\mathbf{V}^I(i, idx) \leftarrow \mathbf{V}^I(i, idx) + 1$;

 end;

$\mathbf{V}^I(i, :) \leftarrow \text{Normalize}(\mathbf{V}^I(i, :))$;

end;

$\mathbf{R} \leftarrow \text{SVM}(\mathbf{L}, \mathbf{V}^I)$

3.1 Experimental Setting

We compare the proposed hashing algorithm SSKDH against four state-of-the-art supervised hashing algorithms: semi-supervised hashing (SSH) [15], kernel supervised hashing (KSH) [7], supervised discrete hashing (SDH) [12] and kernel supervised discrete hashing (KSDH) [13]. After encoding each cell into a set of binary codes, we utilize HCC to classify images. For better comparison, we show the results of support vector machine (SVM) with the major voting (MV). However, we do not display the results of hashing algorithms with MV, because this strategy performs very badly due to the small number of training samples. Moreover, we present the baseline results obtained by directly applying SVM and nearest neighbors (NN) to the holistic high-dimensional features extracted from the whole image. Specifically, we first detect scale-invariant keypoints from the whole pathology image and then employ SIFT [8] and HOG [2] to extract features around these keypoints. Next, both descriptors are encoded as 2000-dimensional histograms using the bag-of-words (BoW) method [1]. For parameters η and ϵ in SSKDH, we empirically set $\eta = 10^{-4}$ and $\epsilon = 10^{-4}$; For SVM, we adopt the ℓ_2 -regularized logistic regression in LIBLINEAR [3] and utilize 5-fold cross validation to obtain the best parameters. To construct kernelized data, we apply K-means to labeled training cells to generate 500 anchors. All hashing algorithms adopt the same kernel type as KSH.

We randomly partition the lung cancer image dataset into two parts: a training set with 210 adenocarcinoma and 230 squamous images and a testing set containing 800 images, which totally consists of around 1198K cells. Then we randomly select 5 images of each class from the training set as labeled ones, and the remaining training images are regarded as unlabeled ones. Next, we randomly select 200 cells from each training image. Hence, there are totally 2K labeled and 86K unlabeled cells. We repeat above process 20 times and calculate the average accuracy.

3.2 Results and Analysis

Table 1 displays the classification accuracy of various algorithms on test images, and their time including training and testing. Note that, training time for SVM, which utilizes features extracted from the whole image, measures only the model parameter learning and does not include the generation of the high-dimensional BoW vector; similarly, training time of the algorithms using cell features does not contain the cost of cell detection and segmentation. Testing time corresponds to the classification of one test image. As we can see, SVM using cell features and major voting to classify images can achieve superior performance to that using features directly extracted from the whole image. It illustrates the strength of cellular features. In addition, hashing algorithms except SSH with HCC obtain higher accuracy than SVM+MV. Among all hashing algorithms, SSKDH attains the best accuracy, the gain is 1.33% and 2.82% over the best competitor on HOG and GIST features, respectively. The main reason is that SSKDH can utilize both semantic (high-level) information from labeled cells and the low-level features from unlabeled ones.

Table 1. Image classification accuracy (%) of different methods with 5 labeled images of each class on 800 test images. (Hashing algorithms encode each cell into 4-bit binary codes. **Image** means directly extracting the holistic high-dimensional features from the whole image and **Cell** means extracting features from cells.)

Method	Adeno	Squam	Mean	Adeno	Squam	Mean	Time	
Image	HOG			SIFT			Train	Test
SVM	64.13	50.88	57.40	74.38	58.56	66.47	0.01	3.00×10^{-4}
NN	82.15	31.84	56.59	75.04	54.88	64.96	-	1.87×10^{-4}
Cell	HOG			GIST			Train	Test
SVM + MV	66.50	97.50	82.00	92.75	66.50	79.63	0.58	1.84×10^{-1}
SSH + HCC	99.25	58.25	78.75	94.75	42.75	68.75	1.69	3.05×10^{-2}
KSH + HCC	76.25	91.25	83.75	73.26	95.00	84.13	39.41	3.05×10^{-2}
SDH + HCC	78.46	95.00	86.73	84.75	84.25	84.50	17.28	3.05×10^{-2}
KSDH + HCC	71.00	98.00	84.50	71.76	92.00	81.88	17.17	3.05×10^{-2}
SSKDH + HCC	82.50	93.25	87.88	82.01	91.75	86.88	20.05	3.05×10^{-2}

To better show the strength of the proposed framework, especially SSKDH, Fig. 2 shows the performance of hashing algorithms on different number of bits from 2 to 8. It further demonstrates the effectiveness and efficiency of SSKDH and HCC. Note that we set the maximum number of bits to 8 in Fig. 2, because each training image only uses 200 cells, and sometimes more bits would decrease the performance of hashing algorithms with HCC. Additionally, all hashing algorithms can attain the best or sub-best performance at 2-bit, because these images only contains two diseases leading to $rank(\mathbf{S}) = 2$.

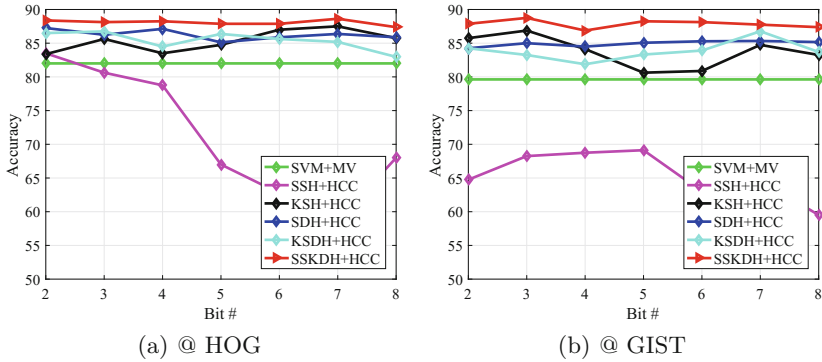


Fig. 2. Classification accuracy vs the number of bits. (a) Cells represented by HOG features; (b) cells represented by GIST features

4 Conclusion

In this paper, we design a novel cell-based framework using few labeled images for pathology image classification. Firstly, we propose a novel semi-supervised hashing model, which can utilize the cell information in both labeled and unlabeled images, to encode each cell into binary codes. Then we propose a new image encoding method, namely hashing codes based classification, to map binary codes of cells in one whole image into a relatively high-dimensional vector. Finally, we use these vectors and learn an SVM for image classification. Extensive experiments on the lung cancer image dataset demonstrate the effectiveness and efficiency of the proposed framework.

References

1. Caicedo, J.C., Cruz, A., Gonzalez, F.A.: Histopathology image classification using bag of features and kernel functions. In: Combi, C., Shahar, Y., Abu-Hanna, A. (eds.) AIME 2009. LNCS, vol. 5651, pp. 126–135. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-02976-9_17](https://doi.org/10.1007/978-3-642-02976-9_17)
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, pp. 886–893 (2005)
3. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: a library for large linear classification. *J. Mach. Learn. Res.* **9**(Aug), 1871–1874 (2008)
4. Hou, L., Samaras, D., Kurc, T.M., Gao, Y., Davis, J.E., Saltz, J.H.: Patch-based convolutional neural network for whole slide tissue image classification. In: CVPR, pp. 2424–2433 (2016)
5. National Cancer Institute: The Cancer Genome Atlas (2013). <https://tcga-data.nci.nih.gov>
6. Jiang, M., Zhang, S., Huang, J., Yang, L., Metaxas, D.N.: Joint kernel-based supervised hashing for scalable histopathological image analysis. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 366–373. Springer, Cham (2015). doi:[10.1007/978-3-319-24574-4_44](https://doi.org/10.1007/978-3-319-24574-4_44)

7. Liu, W., Wang, J., Ji, R., Jiang, Y., Chang, S.: Supervised hashing with kernels. In: CVPR, pp. 2074–2081 (2012)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
9. Mesbah, S., Conjeti, S., Kumaraswamy, A., Rautenberg, P., Navab, N., Katouzian, A.: Hashing forests for morphological search and retrieval in neuroscientific image databases. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9350, pp. 135–143. Springer, Cham (2015). doi:[10.1007/978-3-319-24571-3_17](https://doi.org/10.1007/978-3-319-24571-3_17)
10. Petushi, S., Garcia, F.U., Haber, M.M., Katsinis, C., Tozeren, A.: Large-scale computations on histology images reveal grade-differentiating parameters for breast cancer. *Bio. Med. Comput. Med. Imag.* **6**(1), 1 (2006)
11. Shah, A., Conjeti, S., Navab, N., Katouzian, A.: Deeply learnt hashing forests for content based image retrieval in prostate MR images. In: SPIE Medical Imaging, p. 978414 (2016)
12. Shen, F., Shen, C., Liu, W., Shen, H.: Supervised discrete hashing. In: CVPR, pp. 37–45 (2015)
13. Shi, X., Xing, F., Cai, J., Zhang, Z., Xie, Y., Yang, L.: Kernel-based supervised discrete hashing for image retrieval. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 419–433. Springer, Cham (2016). doi:[10.1007/978-3-319-46478-7_26](https://doi.org/10.1007/978-3-319-46478-7_26)
14. Shi, X., Xing, F., Xu, K., Sapkota, M., Yang, L.: Asymmetric discrete graph hashing. In: AAAI (2017)
15. Wang, J., Kumar, S., Chang, S.: Semi-supervised hashing for large-scale search. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(12), 2393–2406 (2012)
16. Xing, F., Su, H., Neltner, J., Yang, L.: Automatic Ki-67 counting using robust cell detection and online dictionary learning. *IEEE Trans. Bio. Med. Eng.* **61**(3), 859–870 (2014)
17. Yang, L., Chen, W., Meer, P., Salaru, G., Feldman, M.D., Foran, D.J.: High throughput analysis of breast cancer specimens on the grid. In: Ayache, N., Ourselin, S., Maeder, A. (eds.) MICCAI 2007. LNCS, vol. 4791, pp. 617–625. Springer, Heidelberg (2007). doi:[10.1007/978-3-540-75757-3_75](https://doi.org/10.1007/978-3-540-75757-3_75)
18. Zhang, S., Metaxas, D.: Large-scale medical image analytics: recent methodologies, applications and future directions. *Med. Imag. Anal.* **33**, 98–101 (2016)
19. Zhang, X., Su, H., Yang, L., Zhang, S.: Fine-grained histopathological image analysis via robust segmentation and large-scale retrieval. In: CVPR, pp. 5361–5368 (2015)
20. Zhang, X., Su, H., Yang, L., Zhang, S.: Weighted hashing with multiple cues for cell-level analysis of histopathological images. In: Ourselin, S., Alexander, D.C., Westin, C.-F., Cardoso, M.J. (eds.) IPMI 2015. LNCS, vol. 9123, pp. 303–314. Springer, Cham (2015). doi:[10.1007/978-3-319-19992-4_23](https://doi.org/10.1007/978-3-319-19992-4_23)