Pattern Recognition 51 (2016) 12–26

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

An automated pattern recognition system for classifying indirect immunofluorescence images of HEp-2 cells and specimens

Siyamalan Manivannan, Wenqi Li, Shazia Akbar, Ruixuan Wang, Jianguo Zhang, Stephen J. McKenna*

CVIP, School of Computing, University of Dundee, UK

ARTICLE INFO

Article history:
Received 21 January 2015
Received in revised form
15 July 2015
Accepted 5 September 2015
Available online 28 September 2015

Keywords:
Anti-nuclear antibody test
Cell classification
Subcellular fluorescence patterns
HEp-2 cells
Multi-resolution local patterns
Ensemble SVM

ABSTRACT

Immunofluorescence antinuclear antibody tests are important for diagnosis and management of autoimmune conditions; a key step that would benefit from reliable automation is the recognition of subcellular patterns suggestive of different diseases. We present a system to recognize such patterns, at cellular and specimen levels, in images of HEp-2 cells. Ensembles of SVMs were trained to classify cells into six classes based on sparse encoding of texture features with cell pyramids, capturing spatial, multiscale structure. A similar approach was used to classify specimens into seven classes. Software implementations were submitted to an international contest hosted by ICPR 2014 (Performance Evaluation of Indirect Immunofluorescence Image Analysis Systems). Mean class accuracies obtained on heldout test data sets were 87.1% and 88.5% for cell and specimen classification respectively. These were the highest achieved in the competition, suggesting that our methods are state-of-the-art. We provide detailed descriptions and extensive experiments with various features and encoding methods.

1. Introduction

Antinuclear antibody (ANA) tests are important in the diagnosis and management of autoimmune diseases. These include systemic lupus erythematosus, Sjogren's syndrome, rheumatoid arthritis, polymyositis, scleroderma, Hashimoto's thyroiditis, juvenile diabetes mellitus, Addison disease, vitiligo, pernicious anemia, glomerulonephritis, and pulmonary fibrosis. Immunofluorescene ANA tests have been recommended as the gold standard for ANA testing due to their relatively high sensitivity [1]. Specifically, human epithelial (HEp-2) cell specimens are examined using Indirect ImmunoFluorescence (IIF) imaging [2]. The flow of the IIF procedure includes the following steps: image acquisition, mitosis detection, fluorescence intensity classification and staining pattern recognition. The pattern recognition step is an important one as different patterns are suggestive of different autoimmune diseases. A nucleolar pattern is often associated with scleroderma, for example. Manual analysis of IIF images is laborious and time-consuming. Furthermore, two or more experts can be required to examine each ANA sample due to inter-observer variability. Aiming to standardize the procedure compared to current manual practice and to reduce workloads, computer aided diagnosis (CAD) systems have been proposed for the analysis IIF images [3,4].

This paper describes a system to classify pre-segmented immunofluorescence images of individual HEp-2 cells into six classes (homogeneous, speckled, nucleolar, centromere, golgi, and nuclear membrane) as well as a system to classify HEp-2 specimen images into seven classes (homogeneous, speckled, nucleolar, centromere, golgi, nuclear membrane and mitotic spindle). Instances from these classes are shown in Figs. 1 and 2. These two classification tasks correspond to those used in the contest on *Performance Evaluation of Indirect Immunofluorescence Image Analysis Systems* (*I3A*)[1] held in conjunction with the 22nd International Conference on Pattern Recognition (ICPR 2014). Our approach to classifying cell and specimen images is based on sets of local features which describe texture and intensity properties of local image regions. Extracted features are encoded via sparse coding and classification is performed using support vector machine (SVM) ensembles.

After reviewing competing methods (Section 2) and describing our proposed method in detail (Sections 3–7), results on the I3A

* Corresponding author at: Computing, Queen Mother Building, University of Dundee, Dundee DD1 4HN Tel. +44 (0)1382 384732.
E-mail address: s.j.z.mckenna@dundee.ac.uk (S.J. McKenna).

[1] http://i3a2014.unisa.it

http://dx.doi.org/10.1016/j.patcog.2015.09.015
0031-3203/© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

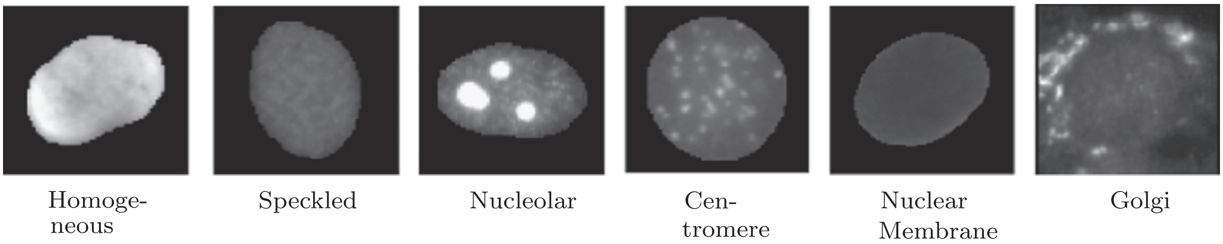

Fig. 1. Sample images from the I3A Task 1 dataset (individual cell classification).

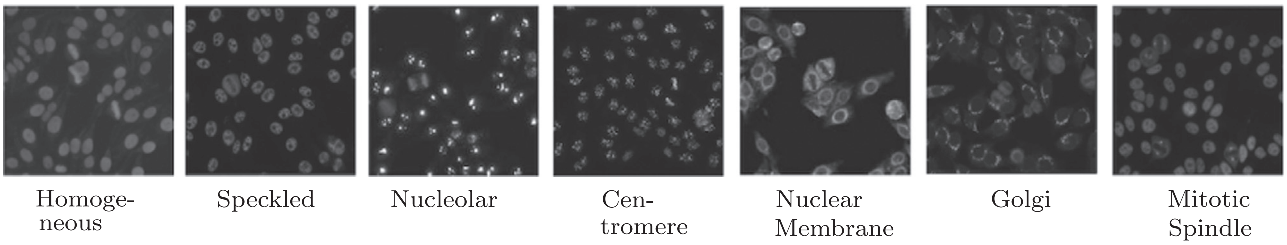

Fig. 2. Sample images from the I3A Task 2 dataset (specimen classification).

contest datasets are presented (Sections 8–10). This paper builds on our earlier I3A workshop papers [5,6]. It sets the proposed method in the context of related literature, describes it in more detail, presents more extensive experiments to investigate the effect of various components on performance, and summarizes performance in direct comparison with other methods. We present systems tailored to capture cell class-specific properties, leveraging state-of-the-art computer vision techniques. The experiments we report should help guide future developments in this area by providing evidence on the relative contributions of, e.g., feature combinations, data augmentation, cell pyramids, and sparse coding. Systems we proposed and describe in this paper won both tasks at the I3A contest in controlled tests. As such they can serve as benchmarks for researchers developing pattern recognition systems for this important application.

2. Related work

This section concisely reviews previous work related to HEp-2 cell image classification in the context of ANA testing. There exists a wider literature on the recognition of fluorescence image patterns characteristic of subcellular structures more generally [7]. However, its review is beyond the scope of this paper.

Perner et al. [8] presented an early attempt at developing an automated HEp-2 cell classification system. Cell regions were represented by a set of basic features extracted from binary images obtained at multiple grey level thresholds. Those features were then classified into six categories by a decision tree algorithm. This approach was further employed and integrated by Sack et al. [9] for identification of positive fluorescence and a set of immunofluorescence patterns. Hsieh et al. [10] performed classification of immunofluorescence patterns using learning vector quantization (LVQ) and various texture features that included grey-level histogram statistics, co-occurrence matrix features, and an estimate of fractal dimension.

The problem of HEp-2 cell classification attracted increased attention among researchers with the benchmarking contests held in conjunction with the International Conference on Pattern Recognition (ICPR) in 2012 [11] (*HEp-2 Cell Classification*[2]) and the International Conference on Image Processing (ICIP) in 2013 [12] (*Competition on Cells Classification by Fluorescent Image Analysis*[3]). A special issue following the ICPR 2012 contest was also organized in the journal Pattern Recognition [4]. Various image descriptors were adopted in those contests, including popular local texture features such as local binary patterns (LBP) and its variants (e.g., multi-resolution LBP), SIFT, summative intensity statistics (e.g., mean and standard deviation) of local or whole image regions, and morphological properties (e.g., eccentricity) [12]. Standard feature encoding methods, in particular bag of words (BoW), were often applied to represent the statistics of local features in the feature space. The majority of the classifiers used were support vector machines although *k*-nearest neighbour classifiers, boosting classifiers, random forest classifiers, and neural networks were also used by some groups [4,12]. For more detailed descriptions of the methods submitted to those contests, the reader is referred to the associated reports [4,11,12].

The I3A contest held in conjunction with ICPR 2014 was the most recent in this series of contests. It received 11 submissions to Task 1 (cell classification) and 7 submissions to Task 2 (specimen classification) [3]. Methods submitted for Task 1 can be classified broadly into two categories: those with feature coding and those without feature coding. The feature coding-based methods basically adopted the popular bag of words framework using either hard coding or sparse coding [13,14] and either SVM or boosting classifiers. They differed in their choice of local features. For instance, Ensafi et al. [15] used SIFT and SURF features with sparse coding and max pooling. Theodorakopoulos et al. [16] combined a set of local features, including LBP and rotation-invariant SIFT, with vectors of locally aggregated descriptors (VLAD) [17]. Paisitkriangkrai et al.'s method (as described in [3]) combined features including region covariance and co-occurrence of adjacent LBPs. Gragnaniello et al. [18] used a recently developed local feature called dense scale-invariant descriptors [19] to characterize cell images. The feature coding methods were among the top performing in the contest [3]. The methods without feature coding basically followed a paradigm of global feature representation with a popular machine learning classifier. For example, Roberts' method (as described in [3]) used a set of wavelet-based features with a SVM classifier. Codrescu [20] learned a neural network with

[2] http://mivia.unisa.it/hep2contest

[3] http://http://nerone.diiie.unisa.it/contest-icip-2013

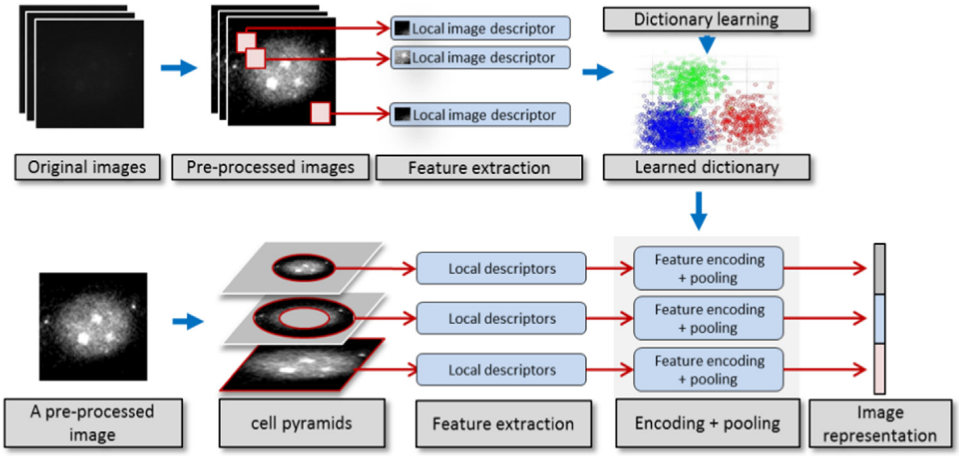

Fig. 3. An overview of the system for generating the image-level feature representation using only one feature type: dictionary learning from training images (first row) and feature encoding to obtain the image-level feature representation (second row). The final image representation is a concatenation of the image level representations obtained by different types of features.

finite impulse response filters applied to cell images. Ponomare et al.'s method (as described in [3]) utilized a set of morphological features (such as number of isolated regions in each cell) with a SVM classifier. Taormina et al.'s method (as described in [3]) used an ensemble of nearest neighbour classifiers on 108 features. Those methods generally performed less well in the contest. It is worth noting that Gao et al. [21] used a deep convolutional neural network (CNN) with 8 layers. The last layer was in principle a logistic regression classifier with a soft-max activation function. This method performed reasonably well in the contest. Although deep CNNs have proven very successful on various large-scale image classification benchmarks, they have very large numbers of parameters and it can be difficult to tune their structure to a specific task such as I3A.

Most of the methods used for Task 2 (specimen classification) first performed classification at the individual cell (or sub-region) level using approaches similar to Task 1 and then aggregated the results by majority voting. For instance, Ensafi et al. [15] performed specimen classification by applying voting to the cell classification results for a small number of extracted cells. Similarly, Liu et al. and Paisitkriangkrai et al. (as described in [3]) classified specimen images based on a majority voting over regions where only features within a cell were considered when describing a region. Ponomare et al. [3] applied an unweighted voting scheme for a final classification of the specimen image based on morphological features.

Notable trends when comparing the I3A contest entries to previous work are the use of more advanced hand-crafted local features (e.g. multi-resolution local patterns with cell pyramids as in our entries [5,6], and dense scale-invariant descriptors [19]), dataset augmentation, and the deployment of deep learning for automatic feature learning. Our proposed method benefits from a combination of the following factors which we believe contribute to its state-of-the-art performance: complementary multi-resolution features, the use of a specifically designed spatial structure for cell images, sparse coding, and data augmentation.

3. System overview

Fig. 3 gives an overview of the system used for generating a feature representation from an image of a cell for input to a classifier. Firstly, each cell image was intensity-normalized. Sets of local features were then extracted and a feature encoding method (e.g. sparse coding) was employed to aggregate the local features into a cell image representation. A two-level cell pyramid was used to capture spatial structure of cell images. Support vector machines were then used to classify images of cells or to classify specimen images containing multiple cells. The following sections describe these system components in detail. Experiments investigating the effect of different system components, feature representations and encodings are then reported.

4. Local feature extraction

Prior to feature extraction, each cell's image was intensity normalized; specifically, the segmentation mask was dilated (using a 5×5 structuring element) and the intensity values within each cell's dilated mask region were then linearly rescaled so that 2% of pixels in each cell became saturated at low and high intensities (Fig. 4).

Local features were extracted densely from each pre-processed cell image. Four types of local feature were considered, namely, Multi-resolution Local Patterns (mLP), Root-SIFT (rSIFT), Random Projections (RP) and Intensity Histograms (IH).

4.1. Multi-resolution Local Patterns

Multi-resolution Local Patterns (mLP) are a multi-resolution adaptation of the local higher-order statistical (LHS) patterns proposed by Sharma et al. [22] for texture classification. LHS is a non-binarized version of the well-known Local Binary Patterns (LBP) descriptor. It operates on a small image neighbourhood of size 3×3. To capture information from a larger neighbourhood and reduce noise effects, we extended LHS from a multi-resolution perspective by employing the sampling patterns described by Mäenpää [23]. This sampling pattern is inspired by the spatial structure of receptive fields in the human retina and has been widely adopted in recently developed visual features in computer vision, e.g., BRISK [24]. Fig. 5 shows an example sampling pattern and the generation of the mLP descriptor. In Fig. 5 the local neighbourhood is quantized radially into three resolutions (radii), and at each resolution a set of ($N=8$) sampling regions (indicated as circles) are considered. At each sampling point a Gaussian filter is applied, integrating information from the filter's region of support. We call the combination of LHS and these sampling patterns *multi-resolution local patterns*.

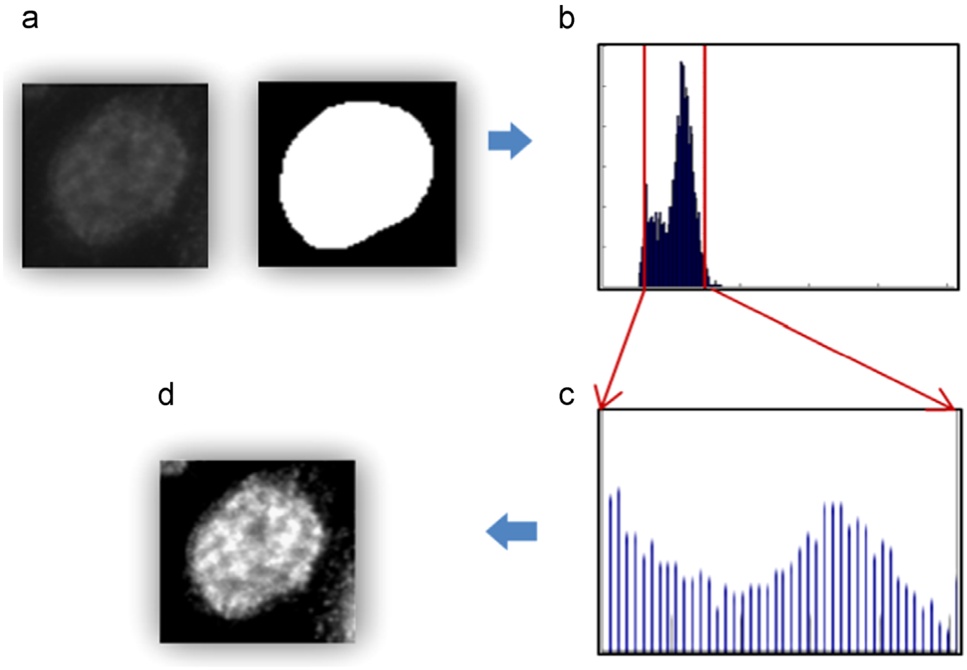

Fig. 4. Image preprocessing: (a) an example cell image and its mask, (b) histogram of intensity values inside cell region in (a), (c) normalized histogram, and (d) preprocessed image.

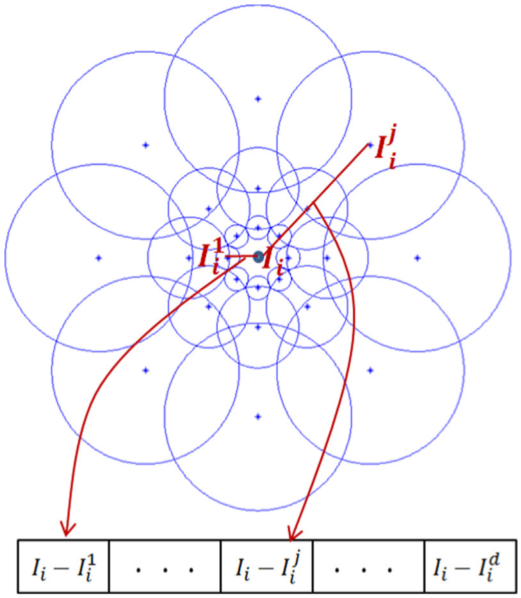

Fig. 5. Generation of multi-resolution local patterns (mLP).

4.2. Root-SIFT

Root-SIFT (rSIFT) is a variant of the widely used SIFT descriptor that produces better performance than SIFT on some image matching and retrieval tasks [25]. The standard SIFT descriptor is a histogram representation of local image derivatives and was originally designed to be used with Euclidean distance. Using Euclidean distance to compare histograms often yields inferior performance compared to other measures such as χ^2 or Hellinger for texture classification and image categorization [25]. Therefore, standard SIFT was modified in [25] to create rSIFT such that comparing rSIFT descriptors using Euclidean distance is equivalent to using the Hellinger kernel to compare SIFT vectors.

4.3. Random projection

Random projection (RP) is a simple yet powerful method for dimensionality reduction [26]. It projects patch intensity vectors from the original patch-vector space $\mathbb{R}^{D'}$ to a compressed space $\mathbb{R}^D$ using randomly chosen projection vectors. Such a scheme has been successfully applied to texture image classification [27]. Let $\mathbf{x}$ be a D'-dimensional patch vector and $\hat{\mathbf{x}}$ be its D-dimensional representation in the compressed space. The RP method simply maps these vectors using a $D \times D'$ random projection matrix R, such that

$$\hat{\mathbf{x}} = R\mathbf{x} \tag{1}$$

Each element in matrix R is sampled from a Gaussian distribution with zero mean and unit variance. The key point of RP is that when projecting the patch-vectors from the original space to the compressed space their relative distances are approximately preserved.

4.4. Intensity histograms

Intensity histograms (IH) were computed from small image patches to represent the local intensity distribution.

5. Feature encoding

Feature encoding methods aggregate the local features from an image or image region and play an important role in classification. Four methods for feature encoding were compared, namely bag-of-words (BoW), a sparse coding method (SC), Fisher vectors (FV), and vectors of locally aggregated descriptors (VLAD). A pyramid was used to encode spatial structure. Each of these methods is now described briefly.

5.1. Bag-of-Words

Bag-of-Words (BoW) is widely applied as a feature encoding method for medical [2] as well as natural [13,28] image classification. In BoW local features sampled from training images are clustered to build a dictionary (codebook). This dictionary represents a set of visual words (or clusters) which are then used to compute a BoW frequency histogram as a feature vector representation of any given image. BoW uses hard quantization where each local image descriptor is assigned to only one visual word.

5.2. Sparse coding

Sparse coding (SC) has shown improved performance over BoW for image classification [28]. In SC each local image descriptor is reconstructed using a sparse weighted combination of visual words. Locality-constrained linear coding (LLC), an efficient variant of sparse coding, utilizes the local linear property of manifolds to project each descriptor into its local coordinate system [14]. Let $X_i \in \mathbb{R}^{D \times N_i}$ be a matrix in which each of the N_i columns is a D-dimensional local descriptor extracted from an image I_i, i.e. $X_i = [\mathbf{x}_{i1}, \mathbf{x}_{i2}, \ldots, \mathbf{x}_{iN_i}]$. Given a codebook with M entries, $B = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_M] \in \mathbb{R}^{D \times M}$, LLC uses the following criterion to compute the codes $C = [\mathbf{c}_{i1}, \mathbf{c}_{i2} \ldots \mathbf{c}_{iN_i}]$:

$$\underset{\mathbf{c}}{\operatorname{argmin}} \sum_{j=1}^{N_i} \| \mathbf{x}_{ij} - B\mathbf{c}_{ij} \|^2 + \lambda \| \mathbf{d}_{ij} \odot \mathbf{c}_{ij} \|^2$$
$$\text{s.t.} \quad \mathbf{1}^T \mathbf{c}_{ij} = 1, \quad \forall j \tag{2}$$

where $\odot$ denotes the element-wise multiplication and

$$\mathbf{d}_{ij} = \exp\left(\frac{\operatorname{dist}(\mathbf{x}_{ij}, B)}{\sigma} \right) \tag{3}$$

where $\operatorname{dist}(\mathbf{x}_{ij}, B) = [\|\mathbf{x}_{ij} - \mathbf{b}_1\|_2^2, \ldots, \|\mathbf{x}_{ij} - \mathbf{b}_M\|_2^2]^T$ and σ is a decay parameter. A fast approximation to LLC was described in [14] to speed up the encoding process. Specifically, instead of solving (2), the $K(<D<M)$ nearest neighbours of $\mathbf{x}_{ij}$ in B were considered as the local bases $\overline{B}_{ij}$ and a much smaller linear system (Eq. (4)) was solved to get the local linear codes:

$$\underset{\mathbf{c}}{\operatorname{argmin}} \sum_{j=1}^{N_i} \| \mathbf{x}_{ij} - \overline{B}_{ij}\mathbf{c}_{ij} \|^2$$
$$\text{s.t.} \quad \mathbf{1}^T \mathbf{c}_{ij} = 1, \quad \forall j \tag{4}$$

The image representation $\mathbf{z}_i$ of an image I_i is then obtained by pooling the sparse codes associated with the local descriptors. Two kinds of pooling, *max* and *sum*, are applied in the literature for SC. The max-pooling can be defined as $z_i^k = \max \mathbf{c}_{ij}^k$, $j = 1, \ldots, N_i$, and the sum pooling can be defined as $z_i^k = \sum_{j=1}^{N_i} |\mathbf{c}_{ij}^k|$, where, $z_i{}^k$ and $\mathbf{c}_{ij}^k$ are respectively the kth element of $\mathbf{z}_i$ and $\mathbf{c}_{ij}$ [29].

5.3. Fisher vectors

Fisher vectors (FV) capture additional information about the distribution of the image descriptors compared to the count (0th-order) statistics in BoW. FV has shown improved performance over BoW and SC for image classification in [30]. In FV, the dictionary is first modelled as a Gaussian mixture model (GMM) $p(\mathbf{x}|\boldsymbol{\Theta})$

$$p(\mathbf{x}|\boldsymbol{\Theta}) = \sum_{m=1}^{M} \pi_m p(\mathbf{x}|\boldsymbol{\mu}_m, \Sigma_m)$$
$$p(\mathbf{x}|\boldsymbol{\mu}_m, \Sigma_m) = \frac{\exp^{-1/2(\mathbf{x}-\boldsymbol{\mu}_m)^T \Sigma_m^{-1}(\mathbf{x}-\boldsymbol{\mu}_m)}}{\sqrt{(2\pi)^d \det(\Sigma_m)}} \tag{5}$$

where $\boldsymbol{\Theta} = (\pi_1, \boldsymbol{\mu}_1, \Sigma_1, \ldots, \pi_M, \boldsymbol{\mu}_M, \Sigma_M)$ are the parameters of the GMM. $\pi_m \in \mathbb{R}^+$ $(\sum_m \pi_m = 1)$, $\boldsymbol{\mu}_m \in \mathbb{R}^D$ and $\Sigma_m \in \mathbb{R}^{D \times D}$ are respectively the weight, the mean and the covariance of the mth Gaussian. GMM uses a soft descriptor-to-cluster assignment:

$$q_m(\mathbf{x}_{ij}) = \frac{\pi_m p(\mathbf{x}_{ij}|\boldsymbol{\mu}_m, \Sigma_m)}{\sum_{l=1}^{M} \pi_l p(\mathbf{x}_{ij}|\boldsymbol{\mu}_l, \Sigma_l)} \tag{6}$$

In FV each cluster is then represented based on the derivative of the GMM with respect to its parameters $\{\boldsymbol{\mu}_m\}$ and $\{\Sigma_m\}$ (1st and 2nd order statistics), i.e.,

$$\mathcal{G}_{\boldsymbol{\mu}_m}^i = \frac{1}{N\sqrt{\pi_m}} \sum_{j=1}^{N_i} q_m(\mathbf{x}_{ij}) \Sigma_m^{-1/2} (\mathbf{x}_{ij} - \boldsymbol{\mu}_m)$$
$$\mathcal{G}_{\boldsymbol{\Sigma}_m}^i = \frac{1}{N\sqrt{2\pi_m}} \sum_{j=1}^{N_i} q_m(\mathbf{x}_{ij}) \left[(\mathbf{x}_{ij} - \boldsymbol{\mu}_m)^T \Sigma_m^{-1} (\mathbf{x}_{ij} - \boldsymbol{\mu}_m) - 1 \right] \tag{7}$$

The final image description $\mathbf{z}_i$ is the concatenation of $\mathcal{G}_{\boldsymbol{\mu}_m}^i$ and $\mathcal{G}_{\boldsymbol{\Sigma}_m}^i$ for all $m = 1, \ldots, M$, leading to a dimensionality of $2MD$.

5.4. Vector of locally aggregated descriptors

Vector of locally aggregated descriptors (VLAD) [17], a simple approximation to FV, uses k-means to learn the dictionary. VLAD uses the 1st-order statistics to represent each cluster; the mth cluster (Q_m) representation of an image I_i can be given as

$$\mathbf{v}_m^i = \sum_{\mathbf{x}_{ij} \in Q_m} \mathbf{x}_{ij} - \boldsymbol{\mu}_m \tag{8}$$

The VLAD image representation $\mathbf{z}_i$ is the concatenation of $\mathbf{v}_m^i$ for all $m = 1, \ldots, M$ followed by L_2 normalization, leading to a dimensionality of MD.

6. HEp-2 cell classification

From each cell image, each of the four feature types was densely extracted from patches of size 12 × 12, 16 × 16, and 20 × 20 pixels with a step-size of 2 pixels.

6.1. SVM ensemble

Augmenting a classifier's training set with rotated versions of the images may improve classification performance but it also increases memory requirements. Instead we used an ensemble of multi-class one-vs-rest, linear SVMs; the ensemble consisted of four SVMs, one trained on the original training set images, and others trained on images after they were rotated through 90°, 180°, and 270°. The overall system which includes data augmentation as well as the ensemble training is shown in Fig. 6.

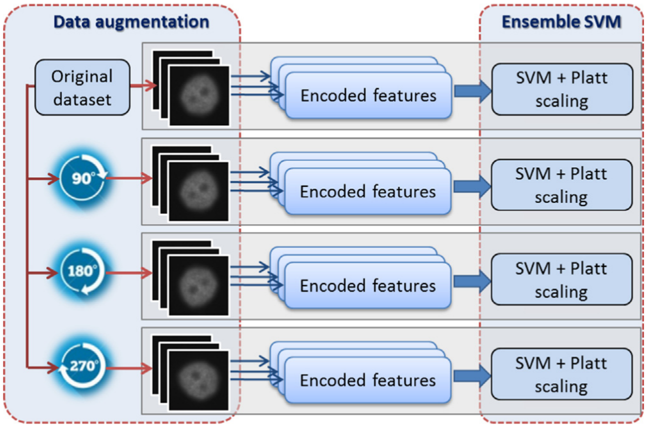

Fig. 6. An overview of the system for data augmentation and SVM ensemble training. Each image can be encoded as shown in Fig. 3.

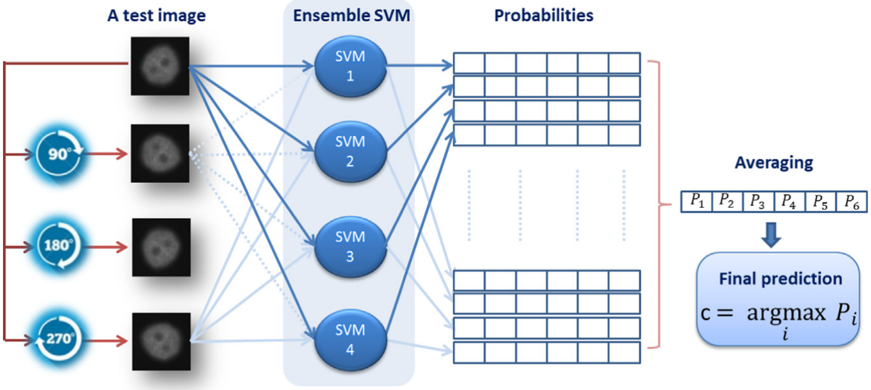

Fig. 7. Testing an image using the SVM ensemble for single cell classification.

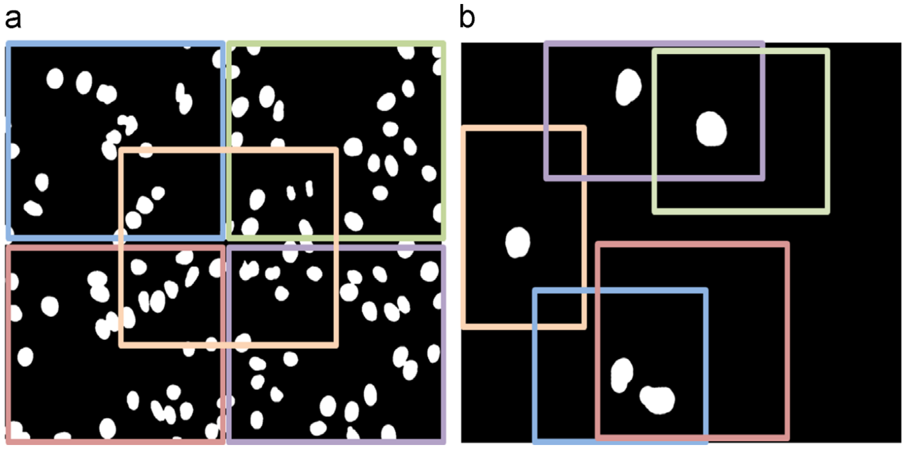

Fig. 8. Sub-images extracted from specimen images for (a) the homogeneous, speckled, nucleolar, centromere, Golgi, and nuclear membrane classes, and (b) the mitotic spindle class. White blobs in the images indicate (a) the segmented regions of individual cells and (b) the manually identified metaphase cells.

At test time, each test image was rotated by 0°, 90°, 180°, and 270°, and each rotated image was then given to the ensemble. This resulted in a set of 16 classification scores for each class (4 rotations × 4 SVMs in the ensemble). Scores were treated as probabilities using Platt rescaling [31]. The final classification decision was made by averaging these probabilistic scores and selecting the highest scoring class. Fig. 7 illustrates the process of classifying a cell image in detail.

Ensemble classification has previously been used for HEp-2 cell classification by Schaefer et al. [32] in the form of a trained linear fusion of classifier outputs. Here the approach we have adopted is based on simply averaging classifier outputs, avoiding the need for training of a fuser. This approach has been used for example to combine the outputs of neural network columns [33] and was shown to perform better than a trained linear combination on handwritten digit recognition [34]. For earlier work on this issue, see e.g., Duin [35].

6.2. Cell pyramids

To capture spatial structure within a cell, a 2-level cell pyramid was used in a similar fashion to the dual-region used by Wiliem et. al. [2,36]. Separate dictionaries of size M were learned for each feature type described above. The encoded local features using these dictionaries were pooled to get an image representation. At the first level of the cell pyramid, the encoded features from the whole cell were pooled to get a feature vector of size P (e.g., for BoW, $P=M$). At the second level, feature vectors were computed from the inner region and from the border region of each cell (see Fig. 3). These three feature vectors were concatenated to give a $3P$-dimensional vector. Finally, encoded features from each of the four feature types were concatenated to give a $12P$-dimensional vector on which classification was based.

7. HEp-2 specimen classification

Our approach for classifying specimens is similar to the approach we proposed for cell image classification, the main difference being that we extracted features at two sets of scales. Specifically, after preprocessing, two types of local feature, mLP and rSIFT, were extracted. These features were densely extracted from image patches using a patch step-size of 4 pixels. Both *small* patches (12 × 12 pixels and 16 × 16 pixels) and *large* patches (48 × 48 pixels and 64 × 64 pixels) were used. Intuitively, small patches can capture local properties at cellular level while large patches can capture information about groups of cells. Features from outside the dilated cell masks were discarded.

A separate dictionary of size M was learned for each feature type with each group of patches (i.e., one for *small* and one for *large* patches). The image representations using the small-patch dictionary and the large-patch dictionary for each feature type were concatenated. Sparse coding with max-pooling was used.

The dataset provided for Task 2, described in Section 8.2, was augmented by including a 90°-rotated version of each original image. This resulted in a set of 2016 images. Five sub-images were extracted from each image based on the layout shown in Fig. 8(a) with the exception of images in the mitotic spindle class.

In mitotic spindle images, metaphase cells in which stained mitotic spindle was apparent were manually identified. Five sub-images were then extracted around those cells, with some random variation, as shown in Fig. 8(b). Finally, the 10,080 extracted sub-images were added to the 2016 images, resulting in an augmented dataset of 12,096 images.

A one-vs-rest, multi-class SVM classifier was then trained on the augmented dataset. Since each specimen was imaged at four different locations in the test SVM classifier was applied to each of these four images, resulting in four sets of classification scores per specimen. Scores were treated as probabilities using Platt rescaling [31]. The final classification decision was made by averaging these probabilistic scores and selecting the class with the highest average score.

8. Experiment setup

8.1. Implementation details

The following parameter settings were used for different feature types:

- mLP: a 3-resolution version with 8 sampling points at each resolution was used as shown in Fig. 5. The parameters of the Gaussian filters at each sampling point were selected as in [23].
- RP: The dimension D' of each linearized patch was reduced to $D=300$ whenever $D' > 300$.
- IH: Local intensity histograms of 256 bins were used.

The public library, `vlfeat` [37], was used for dictionary learning and feature encoding. For SC, we used the implementation of LLC from [14] with 10 nearest neighbours ($K=10$). K-means with 300,000 randomly selected instances of each type of local feature was used to build the dictionaries for BOW, SC and VLAD methods.

In all the reported experiments we used the L_2 and power normalizations [30] to normalize the final image representation $\mathbf{z}_i$ of an image I_i, given by

$$\mathbf{z}_i \leftarrow \frac{\text{sign}(\mathbf{z}_i)|\mathbf{z}_i|^{1/2}}{\|\mathbf{z}_i\|_2} \tag{9}$$

where $|\mathbf{z}_i|^{1/2}$ applies the square root to each component of $\mathbf{z}_i$.

We used the LIBLINEAR [38] implementation for the SVM classifiers. The code[4] from the authors of [39] was used for Platt scaling. We sampled equal number of positive and negative instances from the training set when learning the Platt calibration [31].

Mean Class Accuracy (MCA) was used as one of the evaluation metrics, as required metric by the I3A contest. It is defined as

$$\text{MCA} = \frac{1}{K}\sum_{k=1}^{K} \text{CCR}_k \tag{10}$$

where CCR_k is the correct classification rate for class k and K is the number of classes.

8.2. Datasets

In reported experiments, the Task 1 and Task 2 datasets from the I3A contest were used. These were collected between 2011 and 2013 at the Sullivan Nicolaides pathology laboratory, Australia. For each task, a set of training images was provided to the contest participants. Submitted systems were then evaluated on a separate hidden test set which was privately maintained by the contest organizers and not released to the participants. The results obtained on the contest's hidden test set by our entries are reported in Sections 9.9 and 10.4. This paper also reports cross-validation results on the contest training sets.

The Task 1 dataset consists of 68,429 images of individual cells extracted from 419 patient positive sera (approximately 100–200 cell images per patient serum) along with their binary segmentation masks. 13,596 images were available during training. The remaining 54,833 images were used for the hidden test set to evaluate performance of systems submitted to the contest. The specimens were automatically photographed using a monochrome high dynamic range cooled microscopy camera. Cell images are approximately 70 × 70 pixels in size. The dataset has six pattern classes: homogeneous, speckled, nucleolar, centromere, nuclear membrane, and golgi. An example image from each of the six classes is given in Fig. 1.

The Task 2 dataset consists of uncompressed, monochromatic images of 1001 patient sera with positive ANA test. Each specimen was photographed at four different locations (four images per specimen). A total of 1008 images from 252 specimens were made available (approximately 25% of the data) while the remaining images were retained by the organizers for testing. Each image was 1388 × 1040 pixels and cell masks were obtained based on an automatic segmentation for each image. The dataset has seven pattern classes: homogeneous, speckled, nucleolar, centromere, nuclear membrane, Golgi and mitotic spindle. An example image from each of the seven classes is given in Fig. 2.

The distribution of classes for both tasks is shown in Table 1. The homogeneous, speckled, nucleolar and centromere classes represent common ANA patterns whilst the other three classes are less common.

Experimental results are presented in the following two sections. For cell classification (Section 9), we first used the I3A contest training set to explore the effect that different aspects of the system had on performance. These aspects included feature types, encoding methods, cell pyramids, and data augmentation. Subsequently, we used a leave-one-specimen-out setting to test the ability to classify cells from previously unseen specimens. Finally, we include the results reported on the held-out contest test data for the proposed system as well as for the other contest entries (Section 9.9). For specimen classification (Section 10), we report results comparing performance using different features. We then report results on cross-specimen and cross-image generalization. Finally, we include the results reported on the held-out contest test data (Section 10.4).

9. Cell classification results (Task 1)

9.1. Experiment 1: comparison of different features and encoding methods

We compared the performance of different features and encoding methods on the Task 1 training dataset. Two-fold cross-

Table 1
Distribution of classes in Task 1 and Task 2 training datasets.

Class	Task 1	Task 2
Homogeneous	2494	212
Speckled	2831	208
Nucleolar	2598	200
Centromere	2741	204
Nuclear membrane	2208	84
Golgi	724	40
Mitotic Spindle	–	60

[4] http://www.work.caltech.edu/htlin/program/libsvm/doc/platt.m

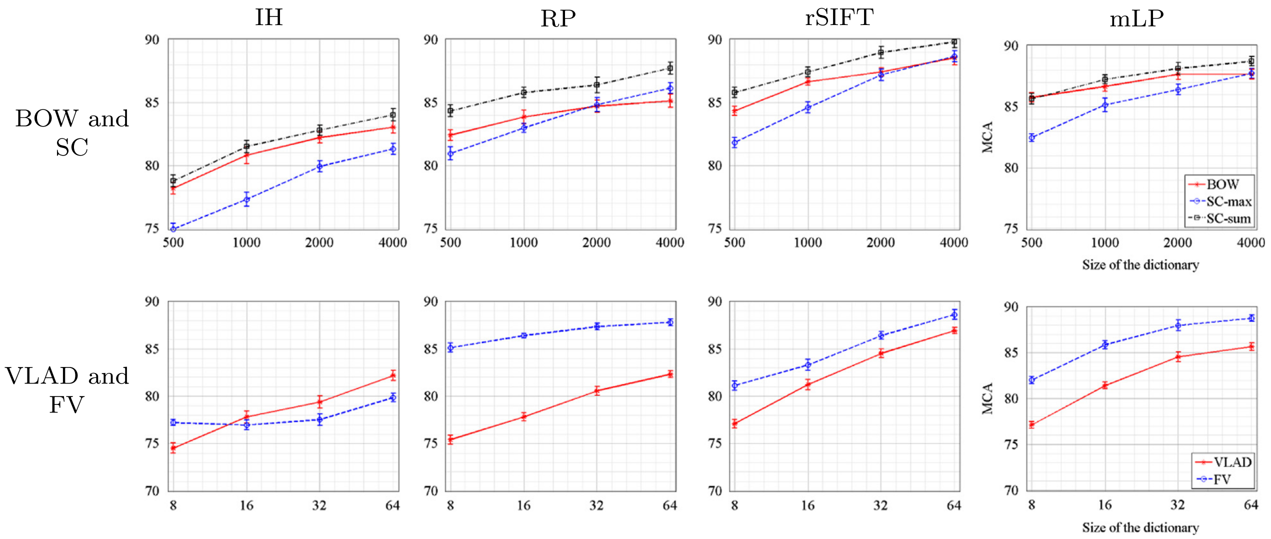

Fig. 9. Performance of various features with different encodings (size of the dictionary vs MCA).

Table 2
Two-fold cross-validation results (MCA ± std) for different feature combinations with and without CPM and data augmentation (dictionary size of 1500).

Features	Original dataset without CPM			Original dataset with CPM			Augmented dataset with CPM		
	BOW	SC-sum	SC-max	BOW	SC-sum	SC-max	BOW	SC-sum	SC-max
rSIFT + mLP	90.4 ± 0.4	91.0 ± 0.4	90.2 ± 0.4	91.1 ± 0.4	92.0 ± 0.5	91.9 ± 0.4	93.6 ± 0.4	94.1 ± 0.3	94.0 ± 0.5
rSIFT + RP	89.6 ± 0.3	90.6 ± 0.3	89.7 ± 0.4	90.6 ± 0.4	91.9 ± 0.4	91.6 ± 0.3	93.1 ± 0.3	93.9 ± 0.3	93.7 ± 0.3
rSIFT + IH	91.0 ± 0.4	91.2 ± 0.3	89.9 ± 0.4	92.6 ± 0.4	93.2 ± 0.3	92.7 ± 0.4	94.2 ± 0.3	94.3 ± 0.3	94.1 ± 0.3
all	92.6 ± 0.3	93.1 ± 0.5	92.6 ± 0.4	93.6 ± 0.4	94.1 ± 0.4	94.1 ± 0.4	95.2 ± 0.3	95.2 ± 0.2	95.2 ± 0.2

Table 3
Different performance measures for classification based on 2-fold cross-validation (all features, SC, max pooling, dictionary size of 1500).

Method	MCA	Accuracy	Precision	Recall	Fscore
Original dataset without CPM	92.6 ± 0.4	97.7 ± 0.1	0.933 ± 0.003	0.925 ± 0.004	0.929 ± 0.003
Original dataset with CPM	94.1 ± 0.4	98.1 ± 0.1	0.941 ± 0.003	0.944 ± 0.003	0.942 ± 0.002
Augmented dataset with CPM	**95.2 ± 0.2**	**98.4 ± 0.1**	**0.949 ± 0.003**	**0.952 ± 0.003**	**0.950 ± 0.003**

validation experiments were carried out, and each was repeated 10 times. Fig. 9 reports the MCAs for different dictionary sizes. rSIFT gave a slightly better performance than the other features. IH gave the worst results. For all encoding methods, larger dictionaries gave higher MCA. SC with sum pooling always gave better performance than other encoding and pooling methods. For all features except IH, FV performed better than VLAD indicating that the additional (2nd order) information it captured was useful. When the dictionary size was 64, FV obtained similar MCA to SC with sum pooling with a dictionary size of 4000, but with an increased feature dimensionality. For example, using rSIFT the dimensionality of an FV image representation was 16,384 compared to 4000 using SC with sum pooling.

9.2. *Experiment 2: combinations of features*

We investigated the performance of combinations of different features. We used BoW and SC encodings for this purpose as they gave better performance than VLAD and FV in Experiment 1. The dictionary size was fixed to 1500. Table 2 reports the results (see columns 2–4). Similar performance was observed using BoW and SC when combining all four types of feature. An improvement of more than 3% was obtained when combining other features with rSIFT (Fig. 9 and Table 2 columns 2–4). Table 5 reports the confusion matrix obtained when combining all the features and encoding with SC and max-pooling. The Golgi class was the least accurately classified; about 8% of Golgi images were misclassified as nucleolar.

9.3. *Experiment 3: effect of cell pyramids*

To improve classification accuracy, particularly of the Golgi class, we incorporated spatial structure into the feature encoding process via cell pyramids (CPM). Table 2 reports the performance of different feature combinations *with* and *without* CPM using BoW and SC approaches (see columns 5–7). When combining all the features and using CPM, the overall MCA was improved by about 1%. In particular, CPM improves the classification accuracy of the Golgi images by about 3% (see Tables 5 and 6).

9.4. *Experiment 4: effect of data augmentation*

We investigated the effect of augmenting the training set by including rotated images as explained in Section 6. An ensemble SVM was used for classification. Augmenting the dataset improved the classification accuracy (see Table 2 columns 8–10 vs. columns 5–7, and Table 7 vs. Table 6). When combining all the features, the overall MCA was further improved by about 1%.

Table 4
Computational time (in sec. averaged over 500 cell images) required for different descriptors for feature extraction and encoding (SC, max pooling, dictionary size of 1500).

Features	Original dataset without CPM			Original dataset with CPM			Augmented dataset with CPM		
	Feature extract	Feature encode	Total	Feature extract	Feature encode	Total	Feature extract	Feature encode	Total
mLP	0.02	0.47	0.49	0.02	0.88	0.91	0.10	3.54	3.64
IH	0.79	0.54	1.33	0.79	1.06	1.85	3.16	4.15	7.31
rSIFT	0.06	0.55	0.61	0.06	1.02	1.08	0.23	4.19	4.42
RP	0.55	0.49	1.04	0.54	0.93	1.46	2.16	3.72	5.88

Table 5
Confusion matrix obtained using all features combined, SC with max pooling, and dictionary size of 1500. (Neither CPM nor data augmentation were used here.)

	Homo.	Spec.	Nucl.	Cent.	NuMe.	Golgi
Homo.	93.5	05.1	00.3	00.1	00.8	00.1
Spec.	04.6	90.6	01.7	02.2	00.6	00.3
Nucl.	01.0	02.1	94.8	01.0	00.6	00.5
Cent.	00.2	03.4	01.6	94.5	00.1	00.1
NuMe.	02.2	01.1	00.8	00.1	95.1	00.6
Golgi.	01.3	01.3	07.7	01.3	01.5	87.0

Table 6
Confusion matrix obtained using all features combined, SC with max pooling, dictionary size of 1500, and CPM. (No data augmentation was used here.)

Homo.	Spec.	Nucl.	Cent.	NuMe.	Golgi
94.7	04.2	00.3	00.1	00.6	00.1
04.0	91.9	01.4	01.7	00.8	00.2
00.9	01.7	95.5	00.9	00.6	00.5
00.1	02.8	01.5	95.5	00.0	00.1
01.9	00.8	00.6	00.1	96.0	00.6
00.9	00.8	05.2	00.6	01.7	90.9

Table 7
Confusion matrix obtained using all features combined, SC with max pooling, dictionary size of 1500, CPM, and data augmentation.

Homo.	Spec.	Nucl.	Cent.	NuMe.	Golgi
95.5	03.3	00.3	00.1	00.5	00.3
04.1	92.1	01.3	01.4	00.8	00.4
00.8	01.2	96.2	00.5	00.5	00.8
00.1	02.3	01.5	96.0	00.0	00.1
01.7	00.4	00.6	00.1	96.5	00.8
00.4	00.3	03.1	00.1	01.0	95.0

Table 8
Confusion matrix for leave-one-specimen-out experiment. (All features, CPM, data augmentation, SC, max pooling, dictionary size of 1500.)

Homo.	Spec.	Nucl.	Cent.	NuMe.	Golgi
81.8	14.8	00.8	00.2	02.0	00.4
09.0	75.5	03.7	10.6	00.8	00.4
01.1	03.4	89.4	02.5	01.3	02.3
00.3	10.7	03.4	85.4	00.0	00.2
05.8	01.9	01.5	00.0	87.9	02.8
04.8	02.1	17.4	01.5	07.5	66.7

BoW and SC gave similar performance; the MCA saturated at about 95%.

9.5. Different measures for classification

In experimental results reported above, MCA was used as the performance measure. This measure was specified for use in the I3A contest. Here we report results using various alternative measures, specifically accuracy, precision, recall, and F-score. Different measures summarize performance in different ways; e.g., accuracy ignores class imbalance thus is biased towards classes that have more examples in the dataset. On the other hand, MCA gives equal importance to all the classes. We direct the interested reader to [40] for detailed explanation and definitions of these measures for multi-class classification. Table 3 reports performance using these measures. These results suggest consistent conclusions regardless of the measure used. Adding CPM improved performance. CPM with data augmentation gave the best performance by all measures.

9.6. Computational time for feature extraction and encoding

Table 4 reports comparisons of the computational time required for feature extraction and encoding in order to compute the cell-level representations. This was by far the most time consuming part of the proposed system. These timings were obtained using Matlab 2014b and Windows 7 running on a machine with a Core i7 processor and 8 GB RAM. IH took more time than other features while resulting in lower MCA (see Fig. 9). On the other hand, mLP took the least time and resulted in competitive MCA. When all feature types were used along with data augmentation and CPM, the system took approximately 21s to compute the cell-level representation for one image.

9.7. Experiment 5: leave-one-specimen-out

The above experiments disregarded the identities of the specimens from which cells had been extracted. To test the generalization performance of our system across different specimens, we conducted an experiment in a *leave-one-specimen-out* setting. Specifically, we used the specimen IDs to split the data into training and validation sets. Since 83 different specimens were available, we used images from 82 specimens for training in each split, and the images from the remaining specimen for testing. In this experiment we used the combination of all feature types, the augmented dataset, CPM, SC, max-pooling, and dictionary size of 1500. Table 8 reports the confusion matrix. An MCA of 81.1% was obtained. The Golgi class had poor results (66.7%). This class exhibits high intra-class variability and was poorly represented in the available data set; only 4 Golgi specimens were in the training set.

9.8. Experiment 6: performance on images extracted from Task 2 dataset

We also made use of cell images segmented from the Task 2 dataset. (We did not use the *mitotic spindle* images in the experiment reported in this section.) An automatic procedure was used to select cells from the Task 2 dataset given the segmentation masks provided with that dataset. Firstly, all disjoint regions were identified in the segmentation mask images using connected component analysis. Secondly, eccentricity values were calculated

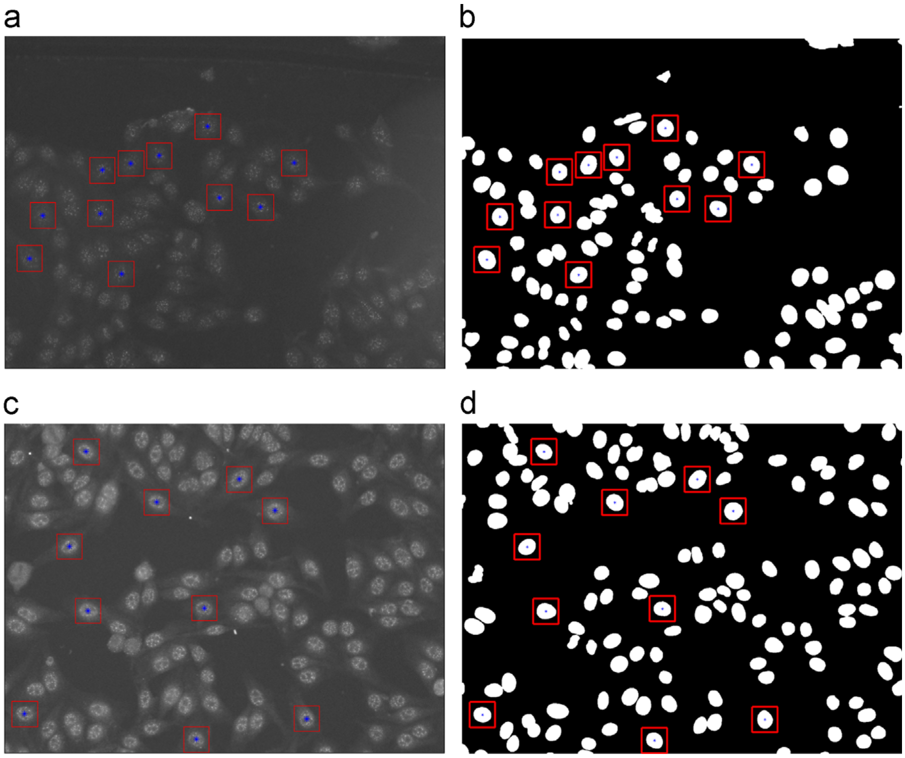

Fig. 10. Sample specimen images from I3A Task-2 dataset. The bounding boxes indicate the cell images which are automatically extracted from these specimen images. (a) A centromere specimen. (b) The segmentation mask for (a). (c) A speckled specimen. (d) The segmentation mask for (c).

for each connected component. Finally, low-eccentricity components that could be bounded by an 80 × 80 square with which no other component overlapped were selected. Approximately 5000 isolated cells were selected in this way. This is illustrated in Fig. 10 where red bounding boxes denote cell images that were extracted.

We trained an ensemble classifier using all the images from the Task 1 training dataset and then tested it on the cell images extracted from the Task 2 dataset. We used the combination of all feature types with the augmented dataset, CPM, SC and max-pooling (dictionary size of 1500). The results are reported in Table 9; an MCA of 86% was obtained.

9.9. Performance on the test dataset

We submitted two systems to the I3A contest for Task 1; the first system used only data made available in the Task 1 training set; the second system trained on a data set consisting of the Task 1 training set together with the additional 5000 cell images extracted from the Task 2 training set (see Section 9.8). Both systems used all the features together with SC (max-pooling, dictionary size 1500), the rotated versions of the images, and CPM.

Fig. 11 reports the MCAs obtained by all of the methods submitted to the contest on the Task 1 test set. Our first submission which made use of only the Task 1 training data obtained an MCA of 84.2%, higher than all the other teams' entries. Our second submission which used additional data (cells extracted from the Task 2 dataset) achieved an MCA of 87.1%. The next best entry, that of Gragnaniello et al. [18], obtained an MCA of 83.6%. Table 10 reports confusion matrices from our method and the method of Gragnaniello et al. The reader is referred to the I3A report [3] for detailed results of other entries.

Table 9
Confusion matrix of the system trained on Task 1 images and tested on the cell images extracted from Task 2 (SC with max pooling, dictionary size of 1500).

	Homo.	Spec.	Nucl.	Cent.	NuMe.	Golgi
Homo.	65.4	28.5	01.5	00.0	03.8	00.7
Spec.	04.5	90.8	00.6	01.7	02.2	00.2
Nucl.	01.2	01.9	95.7	00.0	00.3	00.9
Cent.	00.1	11.1	06.5	82.0	00.2	00.1
NuMe.	03.7	01.9	00.3	00.0	92.0	02.2
Golgi.	00.0	01.4	03.1	00.2	05.1	90.2

9.10. Examples of correctly and incorrectly classified cells

Figs. 12–17 show examples of cells from each class that were correctly and incorrectly classified. Some of the misclassified images are particularly noisy, e.g., the misclassified Golgi images in Fig. 17. In most other cases of misclassification shown, visual inspection reveals qualitative similarity to the class whose label was assigned to it.

10. Specimen classification results (Task 2)

10.1. Experiment 1: comparison of different features

We performed five-fold cross-validation experiments to compare the performance obtained when using different features for Task 2. The dictionary size was fixed to 5000. Table 11 reports the MCA for each feature type as well as their combination. mLP with larger patch sizes outperformed other features. rSIFT with larger patch sizes gave the worst result. Combining features together resulted in an improved MCA of 89.9%. Table 12 shows the confusion matrix.

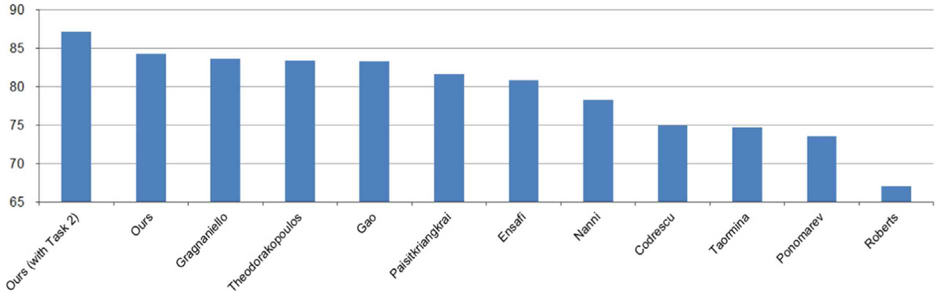

Fig. 11. The MCA at cell level attained by each method on the test set of Task 1.

Table 10
Confusion matrices for the proposed method and that of Gragnaniello et al. [18] on the Task 1 test set.

(a) Proposed method (trained with cell images from Task 1 and Task 2 training sets)

	Cent.	Golgi	Homo.	Nucl.	NuMe.	Spec.
Cent.	97.5	00.1	00.5	00.8	00.2	00.9
Golgi.	00.1	82.0	05.4	03.6	08.2	00.7
Homo.	00.2	00.8	82.6	05.7	04.5	06.1
Nucl.	00.8	00.5	01.4	94.8	01.3	01.3
NuMe.	00.1	00.6	04.9	00.7	92.2	01.4
Spec.	10.3	00.5	12.0	02.0	01.5	73.6

(b) Gragnaniello et al. [18]

	Cent.	Golgi	Homo.	Nucl.	NuMe.	Spec.
Cent.	95.5	00.4	00.2	01.2	00.1	02.7
Golgi.	00.0	71.8	04.7	07.3	14.6	01.6
Homo.	00.1	00.8	78.6	04.9	08.1	07.6
Nucl.	00.8	01.6	02.0	92.5	01.7	01.5
NuMe.	00.1	00.8	03.1	00.9	93.3	01.8
Spec.	13.4	00.7	11.1	02.7	02.2	70.0

Table 11
Classification performance using different features based on five-fold cross-validation. (All=histogram of rSIFT plus histogram of mLP, with *smaller* and *larger* patch sizes.)

Feature type	MCA(%)
mLP $(12 \times 12, 16 \times 16)$	85.63
rSIFT $(12 \times 12, 16 \times 16)$	85.62
mLP $(48 \times 48, 64 \times 64)$	87.34
rSIFT $(48 \times 48, 64 \times 64)$	82.20
All	89.93

10.2. Experiment 2: leave-one-specimen-out experiments

A leave-one-specimen-out experiment was carried out using the specimen IDs provided to split the data into training and validation sets. Since 252 different specimens were available, we used images and their rotated versions from 251 specimens for training in each split. Table 13 reports the confusion matrix. Accuracy of 100% was obtained for the centromere and Golgi classes. The mitotic spindle class had the lowest accuracy being confused mostly with the homogeneous class. An overall MCA of 89.9% was obtained.

Table 12
Confusion matrix based on 5 fold-cross-validation for Task 2 specimen image classification.

	Homo.	Spec.	Nucl.	Cent.	Golg.	Nume.	Mits.
Homo.	86.8	09.4	01.9	00.0	00.0	01.9	00.0
Spec.	01.9	96.1	00.0	00.0	00.0	01.9	00.0
Nucl.	00.0	00.0	98.0	02.0	00.0	00.0	00.0
Cent.	00.0	00.0	00.0	100.0	00.0	00.0	00.0
Golg.	00.0	00.0	00.0	00.0	100.0	00.0	00.0
Nume.	00.0	00.0	00.0	00.0	00.0	95.2	04.8
Mits.	26.7	06.7	00.0	00.0	00.0	13.3	53.3

Table 13
Confusion matrix for leave-one-specimen-out experiment.

	Homo.	Spec.	Nucl.	Cent.	Golg.	Nume.	Mits.
Homo.	88.7	09.4	00.0	00.0	00.0	01.9	00.0
Spec.	03.8	94.2	00.0	00.0	00.0	01.9	00.0
Nucl.	00.0	00.0	98.00	02.00	00.0	00.0	00.0
Cent.	00.0	00.0	00.0	100.00	00.0	00.0	00.0
Golg.	00.0	00.0	00.0	00.0	100.0	00.0	00.0
Nume.	00.0	00.0	00.0	00.0	00.0	95.2	04.7
Mits.	26.7	06.7	00.0	00.0	00.0	13.3	53.3

10.3. Experiment 3: classification of individual images taken at different locations

In the above two experiments (Sections 10.1 and 10.2), the classification decision for each specimen was made based on averaging the classification scores (probabilities) of its four images taken from different locations as explained in Section 7. In this experiment we consider each of the four images separately and classify them individually. A leave-one-specimen-out experiment was performed, where at each iteration a classifier was trained on the images (and their rotated versions) obtained from 251 specimens, and tested on each of the four images of the test specimen. An MCA of 87.9% was obtained, 2% lower than the best accuracy (89.9%) obtained in the experiment reported in Section 10.2.

10.4. Performance on the test dataset

Fig. 18 reports the MCAs obtained by each of the submitted methods on the Task 2 test set. Our method achieved an MCA of 88.5%, outperforming all the other methods submitted. The second placed method, that of Liu et al. (described in [3]), achieved an MCA of 86.1%. Table 14 reports the confusion matrices of our method and that of Liu et al.

11. Conclusion and recommendations

In this paper we explained in detail our winning entries for both Task 1 (cell image classification) and Task 2 (specimen image classification) of the I3A contest. To more fully understand the contributions of the components of our classification systems, we

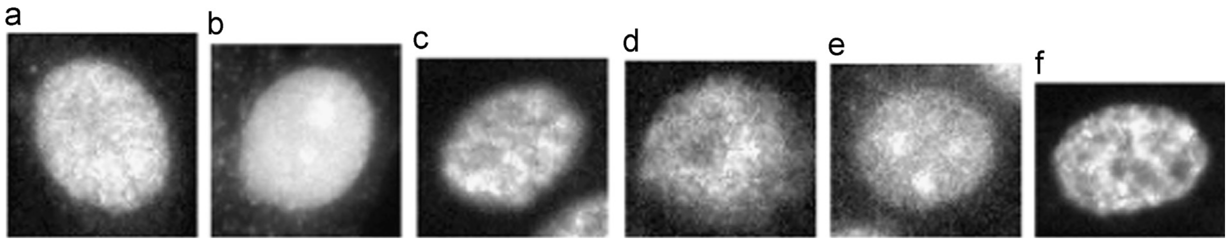

Fig. 12. Examples of *Homogeneous* cells that were (a)–(c) correctly classified, (d) classified as Spec., (e) classified as Nucl., and (f) classified as NuMe.

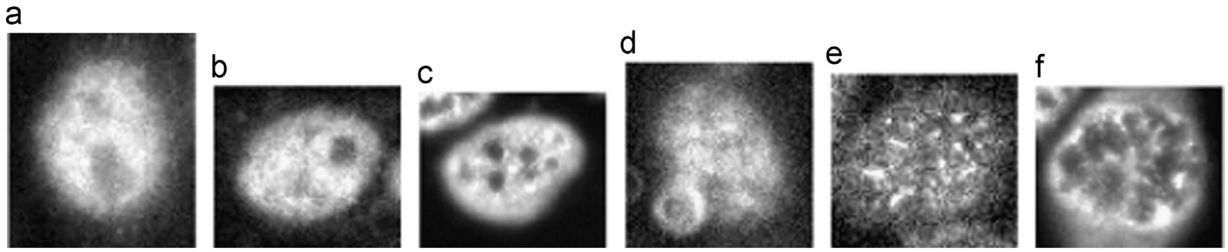

Fig. 13. Examples of *Speckled* cells that were (a)–(c) correctly classified, (d) classified as Homo., (e) classified as Cent., and (f) classified as Golg.

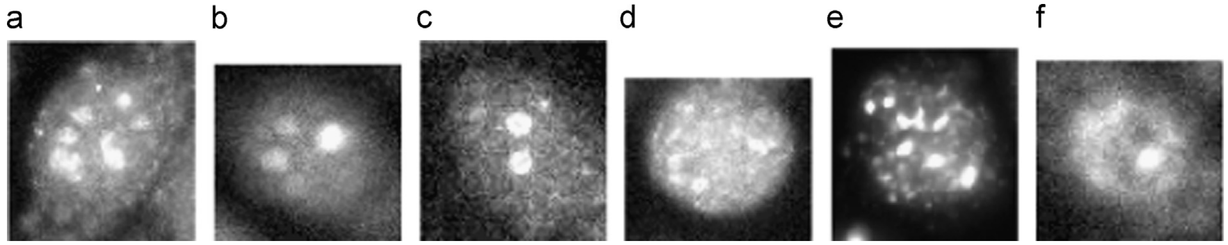

Fig. 14. Examples of *Nucleolar* cells that were (a)–(c) correctly classified, (d) classified as Homo., (e) classified as Cent., and (f) classified as Spec.

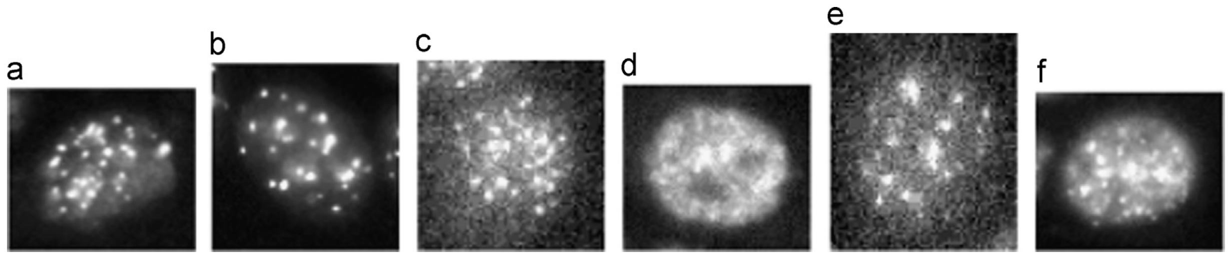

Fig. 15. Examples of *Centromere* cells that were (a)–(c) correctly classified, (d) classified as Homo., (e) classified as Nucl., and (f) classified as Spec.

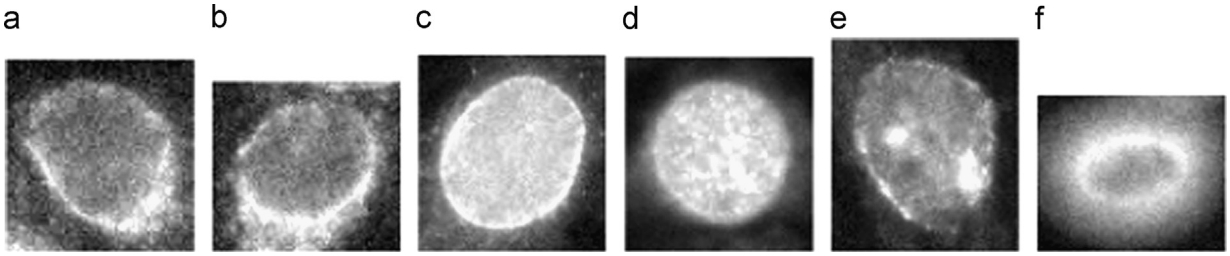

Fig. 16. Examples of *Nuclear Membrane* cells that were (a)–(c) correctly classified, (d) classified as Homo., (e) classified as Nucl., and (f) classified as Spec.

empirically studied the local feature extraction and encoding methods, as well as data augmentations applied in training the systems. We found that (1) the combination of different features outperforms individual feature types; (2) for cell classification SC performs better than BOW. FV and VLAD could achieve similar accuracies to BOW and SC but only with feature representations of much higher dimensionality; (3) adding spatial information from the cell images via the use of cell pyramids can improve classification performance for cell images (an improvement of $\sim 3\%$ was observed for Golgi images); and (4) augmenting the training set by the use of rotated training images further improves the classification performance. When combined, these aspects differentiate

Table 14
Task 2 confusion matrices for (a) our method, and (b) the methods of Liu et. al. (a) Our method (b) Liu et. al..

(a)

	Cent.	Golgi	Homo.	Nucl.	NuMe.	Spec.	MitSp
Cent.	98.7	00.0	00.0	00.7	00.0	00.0	00.7
Golgi.	00.0	80.8	00.0	03.9	03.9	11.5	00.0
Homo.	00.0	00.0	93.0	00.0	00.0	01.9	05.1
Spec.	00.0	00.0	18.2	61.4	02.3	11.4	06.8
Nucl.	00.0	01.3	00.7	00.7	96.0	00.7	00.7
NuMe.	00.0	00.0	01.6	00.0	00.0	98.4	00.0
MitSp	00.0	00.0	07.0	00.6	00.0	01.3	91.1

(b)

	Cent.	Golgi	Homo.	Nucl.	NuMe.	Spec.	MitSp
Cent.	98.7	00.7	00.0	00.0	00.0	00.0	00.7
Golgi.	00.0	80.8	07.7	03.9	03.9	03.9	00.0
Homo.	00.0	00.0	93.0	00.0	00.0	01.3	05.7
Spec.	00.0	02.3	34.1	52.3	02.3	06.8	02.3
Nucl.	00.0	00.0	00.7	00.7	98.0	00.7	00.0
NuMe.	00.0	00.0	06.5	00.0	00.0	91.9	01.6
MitSp	00.0	00.0	11.4	00.6	00.0	00.0	88.0

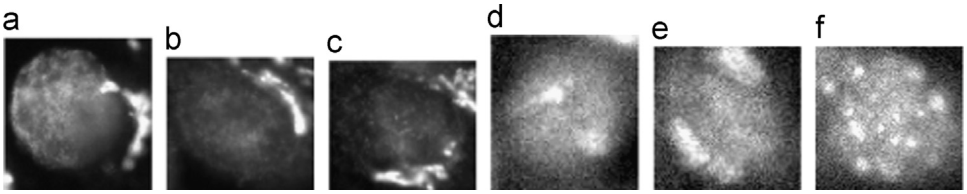

Fig. 17. Examples of *Golgi* cells that were (a)–(c) correctly classified, (d) classified as Nucl., (e) classified as NuMe., and (f) classified as Spec.

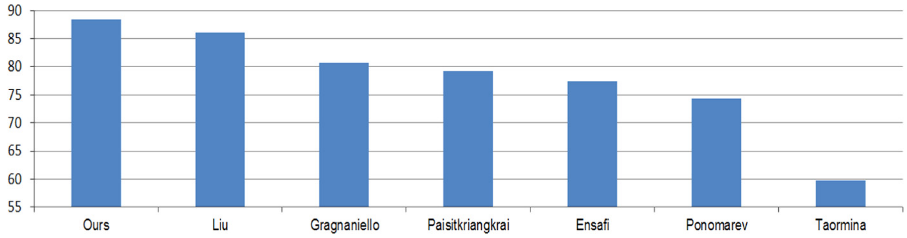

Fig. 18. The MCA at specimen level obtained by each method on the test set of Task 2.

our entries from other entries in the competition. They are important factors to consider in building state-of-the-art cell and specimen image classification systems.

System design choices were guided by using mean class accuracy as the measure of classification performance. (This was the measure specified for use in the I3A contest.) However, experiments with alternative measures suggested similar conclusions. Future contests could be usefully enhanced by investigating the statistical and clinical significance of differences in performance between competing methods. One way to go about assessing statistical significance is bootstrap sampling, as recently incorporated in the PASCAL Visual Object Classes Challenge [41]. It would also be informative to quantify the reliability of the *ground truth* labels, especially given the visual similarity of many of the misclassified examples to their assigned classes (see Figs. 12–17). The features we used were pre-defined (hand-crafted). We believe that future work could further explore learning adaptive feature extractors for cell and specimen classification.

Conflict of interest

None declared.

Acknowledgements

We thank the ICPR-I3A contest organizers, Brian C. Lovell and Arnold Wiliem (University of Queensland, Australia), Gennaro Percannella and Mario Vento (University of Salerno, Italy), for making the data and test results available for this research. S. Manivannan and R. Wang were funded by 2011–2016 EU FP7 ERC project “CODIR: colonic disease investigation by robotic hydrocolonoscopy”, collaborative between the Universities of Dundee (PI Prof. Sir. A. Cuschieri) and Leeds (PI Prof. A. Neville). S. Akbar was funded by the UK Engineering and Physical Sciences Research Council (Grant EP/J500392/1) and W. Li was funded by the overseas student scholarship of the University of Dundee.

References

[1] P.L. Meroni, P.H. Schur, ANA screening: an old test with new recommendations, Ann. Rheum. Dis. 69 (8) (2010) 1402.

[2] A. Wiliem, Y. Wong, C. Sanderson, P. Hobson, S. Chen, B. C. Lovell, Classification of human epithelial type 2 cell indirect immunofluoresence images via codebook based descriptors, in: IEEE Workshop on Applications of Computer Vision (WACV), IEEE, Piscataway, 2013, pp. 95–102.

[3] B.C. Lovell, G. Percannella, M. Vento, A. Wiliem, Performance Evaluation of Indirect Immunofluorescence Image Analysis Systems, Technical Report, ICPR Workshop, 2014.

[4] P. Foggia, G. Percannella, A. Saggese, M. Vento, Pattern recognition in stained HEp-2 cells: where are we now?, Pattern Recognit. 27 (2014) 2305–2314.

[5] S. Manivannan, W. Li, S. Akbar, R. Wang, J. Zhang, S. J. McKenna, HEp-2 cell classification using multi-resolution local patterns and ensemble SVMs, in: 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), IEEE, Piscataway, NJ, 2014, pp. 37–40, http://dx.doi.org/10.1109/I3A.2014.1.

[6] S. Manivannan, W. Li, S. Akbar, R. Wang, J. Zhang, S. J. McKenna, HEp-2 specimen classification using multi-resolution local patterns and svm, in: 2014 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), IEEE, Piscataway, NJ, 2014, pp. 41–44, http://dx.doi.org/10.1109/I3A.2014.20.

[7] X. Chen, M. Velliste, R.F. Murphy, Automated interpretation of subcellular patterns in fluorescence microscope images for location proteomics, Cytom. Part A 69A (2006) 631–640.

[8] P. Perner, H. Perner, B. Muller, Mining knowledge for HEp-2 cell image classification, Artif. Intell. Med. 26 (1–2) (2002) 161–173.

[9] U. Sack, S. Knoechner, H. Warschkau, U. Pigla, F. Emmrich, M. Kamprad, Computer-assisted classification of HEp-2 immunofluorescence patterns in autoimmune diagnostics, Autoimmun. Rev. 2 (5) (2003) 298–304.

[10] T.-Y. Hsieh, Y.-C. Huang, C.-W. Chung, Y.-L. Huang, HEp-2 cell classification in indirect immunofluorescence images, in: 7th International Conference on Information, Communications and Signal Processing, Piscataway, NJ, 2009, pp. 1–4.

[11] P. Foggia, G. Percannella, P. Soda, M. Vento, Benchmarking HEp-2 cells classification methods, IEEE Trans. Med. Imaging 32 (10) (2013) 1878–1889.

[12] P. Hobson, G. Percannella, M. Vento, A. Wiliem, International Competition on Cells Classification by Fluorescent Image Analysis, Technical Report, ICIP, 2013. 〈http://nerone.diiie.unisa.it/contest-icip-2013/ICIP2013_report.pdf〉.

[13] J. Zhang, M. Marszałek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, Int. J. Comput. Vis. 73 (2) (2007) 213–238.

[14] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Piscataway, NJ, 2010, pp. 3360–3367.

[15] S. Ensafi, S. Lu, A. Kassim, C. L. Tan, A bag of words based approach for classification of HEp-2 cell images, in: 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014, pp. 29–32.

[16] I. Theodorakopoulos, D. Kastaniotis, G. Economou, S. Fotopoulos, HEp-2 cells classification using morphological features and a bundle of local gradient descriptors, in: 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014, pp. 33–36.

[17] H. Jégou, M. Douze, C. Schmid, P. Pérez, Aggregating local descriptors into a compact image representation, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 3304–3311.

[18] D. Gragnaniello, C. Sansone, L. Verdoliva, Biologically-inspired dense local descriptor for indirect immunofluorescence image classification, in: 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014, pp. 1–5.

[19] I. Kokkinos, M. Bronstein, A. Yuille, Dense Scale Invariant Descriptors for Images and Surface, Technical Report 7914, INRIA, 2012.

[20] C. Codrescu, Quadratic recurrent finite impulse response MLP for indirect immunofluorescence image recognition, in: 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014, pp. 49–52.

[21] Z. Gao, J. Zhang, L. Zhou, L. Wang, HEp-2 cell image classification with convolutional neural networks, in: 1st Workshop on Pattern Recognition Techniques for Indirect Immunofluorescence Images (I3A), 2014, pp. 24–28.

[22] G. Sharma, S. ul Hussain, F. Jurie, Local higher-order statistics (LHS) for texture categorization and facial analysis, in: European Conference on Computer Vision (ECCV), 2012, pp. 1–12.

[23] T. Mäenpää, The local binary pattern approach to texture analysis: extensions and applications (Ph.D. thesis), University of Oulu, 2003.

[24] S. Leutenegger, M. Chli, R. Y. Siegwart, BRISK: Binary robust invariant scalable keypoints, in: IEEE International Conference on Computer Vision (ICCV), IEEE, Piscataway, NJ, 2011, pp. 2548–2555.

[25] R. Arandjelovic, A. Zisserman, Three things everyone should know to improve object retrieval, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 2911–2918.

[26] E. Bingham, H. Mannila, Random projection in dimensionality reduction: applications to image and text data, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2001, pp. 245–250.

[27] L. Liu, P. Fieguth, Texture classification from random features, IEEE Trans. Pattern Anal. Mach. Intell. 34 (3) (2012) 574–586.

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

[29] Y.-L. Boureau, N. Le Roux, F. Bach, J. Ponce, Y. LeCun, Ask the locals: multi-way local pooling for image recognition, in: IEEE International Conference on Computer Vision (ICCV), 2011, pp. 2651–2658.

[30] F. Perronnin, J. Sánchez, T. Mensink, Improving the fisher kernel for large-scale image classification, Computer Vision—ECCV LNCS, 6316, Special Publication, Crete, Greece, 2010.

[31] J.C. Platt, Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, Advances in large margin classifiers, MIT Press, Cambridge, MA, USA (2000), p. 61–74.

[32] G. Schaefer, N.P. Doshi, B. Krawczyk, HEp-2 cell classification using multidimensional local binary patterns and ensemble classification, in: Second IAPR Asian Conference on Pattern Recognition, 2013, pp. 951–955.

[33] D.C. Ciresan, U. Meier, J. Schmidhuber, Multi-column deep neural networks for image classification, in: CVPR, 2012.

[34] U. Meier, D.C. Ciresan, L.M. Gambardella, J. Schmidhuber, Better digit recognition with a committee of simple neural nets, in: International Conference on Document Analysis and Recognition, 2011, pp. 1135–1139.

[35] R.P.W. Duin, The combining classifier: to train or not to train?, in: International Conference on Pattern Recognition, 2002, pp. 765–770.

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

[37] A. Vedaldi, B. Fulkerson, VLFeat: an open and portable library of computer vision algorithms, in: Proceedings of the International Conference on Multimedia (MM), ACM, New York, NY, USA, 2010, pp. 1469–1472.

[38] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, LIBLINEAR: a library for large linear classification, J. Mach. Learn. Res. 9 (2008) 1871–1874.

[39] H.-T. Lin, C.-J. Lin, R.C. Weng, A note on Platt's probabilistic outputs for support vector machines, Mach. Learn. 68 (3) (2007) 267–276.

[40] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manag. 45 (4) (2009) 427–437.

[41] M. Everingham, S.M. Ali Eslami, L. van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL visual object classes challenge: a retrospective, Int. J. Comput. Vis. 111 (1) (2015) 98–136.

Siyamalan Manivannan received a B.Sc. degree in Computer Science from the University of Jaffna, Sri Lanka in 2006, and then M.Sc. degree in Informatics from the University of Nice, France in 2010. He is currently a Ph.D. student at the University of Dundee, UK. His research interests include pattern recognition and machine learning.

Wenqi Li received a B.Sc. degree in Computer Science from the University of Science and Technology Beijing, Beijing (2010) and an M.Sc. degree in Applied Computing from the University of Dundee (2011). He is now working towards the Ph.D. degree in the School of Computing, University of Dundee.

Shazia Akbar is a postdoctoral fellow at New York University School of Medicine, New York, USA. She received her PhD in 2015 from the University of Dundee. Her research interests include MRI, medical image analysis, feature extraction and pattern recognition.

Ruixuan Wang received B.Eng. and M.Eng. degrees in Automatic Control and Artificial Intelligence from Xi'an Jiaotong University, and a Ph.D. degree in Computer Vision from the National University of Singapore (2007). He was a post-doctoral researcher at University of Dundee from 2007 to 2014 and subsequently at Heriot Watt University. He is currently a research scientist at Toshiba Medical Visualization Systems Europe (TMVSE).

Jianguo Zhang is a Senior Lecturer in the School of Computing, University of Dundee, UK. He received a Ph.D. degree from the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China (2002). His research interests include pattern recognition, computer vision, medical image analysis and machine learning.

Stephen J. McKenna is Chair of Computer Vision and Head of Research at the School of Computing, University of Dundee. He received the B.Sc. degree in Computer Science (University of Edinburgh, 1990) and Ph.D. degree in Medical Image Analysis (University of Dundee, 1994). His interests include biomedical image analysis, computer vision and machine learning.