A new pairwise deep learning feature for environmental microorganism image analysis

Frank Kulwa1 · Chen Li1 · Jinghua Zhang1 · Kimiaki Shirahama2 · Sergey Kosov3 · Xin Zhao1 · Tao Jiang4 · Marcin Grzegorzek5

Received: 10 August 2021 / Accepted: 20 January 2022 / Published online: 7 March 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH, Germany, part of Springer Nature 2022

Abstract

Environmental microorganism (EM) offers a highly efficient, harmless, and low-cost solution to environmental pollution. They are used in sanitation, monitoring, and decomposition of environmental pollutants. However, this depends on the proper identification of suitable microorganisms. In order to fasten, lower the cost, and increase consistency and accuracy of identification, we propose the novel pairwise deep learning features (PDLFs) to analyze microorganisms. The PDLFs technique combines the capability of handcrafted and deep learning features. In this technique, we leverage the Shi and Tomasi interest points by extracting deep learning features from patches which are centered at interest points’ locations. Then, to increase the number of potential features that have intermediate spatial characteristics between nearby interest points, we use Delaunay triangulation theorem and straight line geometric theorem to pair the nearby deep learning features. The potential of pairwise features is justified on the classification of EMs using SVMs, Linear discriminant analysis, Logistic regression, XGBoost and Random Forest classifier. The pairwise features obtain outstanding results of 99.17%, 91.34%, 91.32%, 91.48%, and 99.56%, which are the increase of about 5.95%, 62.40%, 62.37%, 61.84%, and 3.23% in accuracy, F1-score, recall, precision, and specificity respectively, compared to non-paired deep learning features.

Keywords Deep learning features · Feature extraction · Environmental microorganisms · Pairwise features · Image analysis

Introduction

Microorganisms are tiny living organisms which appear as unicellular, multicellular, or acellular (Li et al. 2020). The usage of microorganism gives a low-cost and efficient means for controlling, monitoring and decomposition of pollution (Singh 2017). For instance, Lactobacteria are used for decomposing substances to give nutrients to plants (Zhou et al. 2009), Mnicrothrix parvicella causes bulking in activated sludge (Wang 2014), and Epistyliis is used as an indicator of poor water quality. Thus, deep understanding of microorganisms helps to leverage them.

There are four traditional techniques for identification of microorganisms which include the chemical method that is good, but it leads into secondary pollutants of chemicals (Zhang et al. 2021); the physical technique (which requires expensive equipment) (Li et al. 2019); the molecular biological method that requires expensive equipment (Yamaguchi et al. 2015); and professional researchers. Lastly, is the morphological method, which requires experienced operators to observe microorganism using the microscope and give identification based on shape characteristics (Zhang et al. 2021). This method is time consuming, tedious, inconsistent and biased to operator’s moods. The use of image processing techniques offers a remedy to traditional methods.
Image processing involves six basic steps, which are image acquisition, pre-processing (denoising, segmentation, enhancement, etc.), feature extraction, post-processing (feature selection, feature fusion, feature vector reduction), classification, and evaluation (Li et al. 2019). Good feature techniques are important for better classification results. In response to this, Priya and Srinivasan (2016) use Fourier descriptors as features for the classification of Tuberculosis bacilli from sputum smear using multilayer perceptron neural network, and it achieves outstanding results of 92.5%. The potential colorimetric and histogram features lead excellent performance of 98.19% accuracy on SVM when applied for classification of soil microorganisms (Kruk et al. 2015).

Generally, there are two types of feature extraction techniques, handcrafted and feature learning (Bengio et al. 2013). Handcrafted features are manual features that are extracted based on prior knowledge. Such features are color, texture, geometric features, global shape, and local shape (such as SIFT and SURF). Interest points are very useful in finding handcrafted features, particularly local features (Kosov et al. 2018). Interest points are distinct regions/points used to differentiate between one object (image) to another (Lindeberg 2013). Examples of interest points are corner, blob, and ridge descriptors. They play a significant role in image matching and classification tasks. For instance, SIFT features (which are extracted from corner descriptors of 10 channels of different color modes) are used for image matching of EMs (Lindeberg 2013; Zou et al. 2016; Zieliński et al. 2017). Similarly, Mithra and Emmanuel (2019) apply local SURF and location-oriented histogram features to the deep learning neural network for Tuberculosis bacilli classification. Due to the high feature representation capability of interest points, invariance to rotation, translation, and changes in illumination, moreover, the local image structure around the interest point is rich in local discriminant information content (Schmid et al. 2000). Thus, we take advantage of corner interest points’ locations around the microorganism by extracting deep learning features around them so as to capture the discriminant characteristics of the object of interest (microorganism) on the image.

Feature learning (features) is a high dimension feature formed by the composition of local features. Examples of feature learning are the bag of visual words (BoVW) (Bolovinou et al. 2013), sparse coding (Afzali et al. 2016), and deep learning features. Deep learning features are extracted by deep convolution networks such as ResNet, AlexNet, and VGG-16. They have superior descriptive power than handcrafted feature methods (Kuzovkin et al. 2018), because they mimic the effectiveness of visual cortex in human brain for feature extraction (Hincheny et al. 2007). Thus, they find many applications in microorganism image classification tasks. For example, Kosov et al. (2018) uses a pre-trained VGG-16 for pixel-level deep learning feature extraction from EM images, which are used for classification using a CRF model. Liang et al. (2021) optimizes Inception-V3 (whose fully connected layers are improved using generic algorithm) for feature learning and classification of EMs. Likewise, A 13 layers CNN is used by Luo et al. (2018) in extraction of color channel features from plankton images. VGG-16 is one of the most useful and powerful CNN models in image classification tasks due to its high feature learning capability (Rajaraman et al. 2018; Keçeli et al. 2018). To leverage the superiority of VGG-16, in this study, we use it in the extraction of deep learning features at each detected corner descriptor’s location.

Bag of visual words (BoVW) is among the most popular feature learning techniques because of its robustness and simplicity. However, due to the orderless representation of local features, it does not give optimal performance. In response to that, Savarese et al. (2006) and Zhang and Chen (2009) considered spatial arrangement of features to discover higher order structures for improving the performance of BoVW. Among the techniques of arranging features is by pairing of spatial close visual words where feature descriptors are mapped to the visual words before pairing, then pairing is done on visual words (Ling and Soatto 2007; Lazebnik et al. 2005; Liu et al. 2008). Yet, the pairing of visual words seems to ignore the underlying distribution of pairs of nearby local feature descriptors. As a remedy to that, Morioka and Satoh (2010) and Morioka and Satoh (2011) propose the pairing of spatial close local feature descriptors (SIFT) and treat them as data points in a joint feature space before the building of BoVW. In this way, more improvement is achieved in the object recognition performance on the tested challenging dataset. Inspired by the idea of pairing features, and to the best of our knowledge, no work has been done on the pairing of deep learning features for the classification of microorganism images; therefore, in this study, we pair deep learning features extracted from corner descriptors’ locations and use them for classification of EMs. The main contributions of this paper are:

1. We combine and leverage the capability of interest points (handcrafted features) and deep learning features by extracting deep learning features on image patches which are centered at corner interest points’ locations. The interest points are determined using Shi and Tomasi theorem. This enables the location of distinctive points on the image where local distinctive deep learning features of an object (microorganism) are extracted.
2. Speculating that the middle point between two nearby patches (interest points) have intermediate spatial features, we pair the feature maps of two nearby
interest points. This increases the amount of potential distinctive features of an object. The pairing is achieved using the Delaunay triangular theorem, which concentrates the triangles inside the object of interest, which thus highlights more the object of interest against the noisy background.

The remainder of the paper is organized as follows: “Data and methods” focuses into the data preparation, proposed techniques, and other essential elements of our contributions, whereas “Results and discussion” delves into the experimental results and analysis. The core challenges of the proposed methods in each EM class are discussed in “Discussion”. Finally, in “Conclusion and future work”, the conclusion and future work are underlined.

Data and methods

Data

In this study, we use Environmental Microorganism Dataset 5th Version (EMDS-5) for testing the potential of the pairwise features for the classification task. This dataset is comprised of 21 classes. Each class contains 20 images of the same species. These species include Actinophrys, Arcella, Aspidisca, Codosiga, Colpoda, Epistyris, Euglypha, Paramecium, Rhotera, Vorticella, Noctiluca Ceratium, Stentor, Siprostomum, Keratella Quadrala, Euglena, Gymnodinium, Gonyaulax, Phacus, Stylomchia, and Synchaeta. Some of the EMs images are shown in Fig. 2. For presentation purposes we name these classes as class 1 (c1), class 2 (c2), class 3 (c3)...class 20 (c20), class 21 (c21) respectively. In total there are 420 available images. True corners of the foreground are the most important in this research. To reduce the possibility of false corners, we crop all images that have outer highlighted square frames at the edges of the images and remain with only the true background and foreground. Then all images are resized to 480 × 360 pixel sizes. Example of the microorganisms of three different classes is given in Fig. 1.

Methods

The novel techniques for extracting and pairing of features are elaborated in detail in this section. Additionally, some of the intermediate experimental results are also shown. The main focus is to increase the potential of the deep learning features for the classification of microorganisms. The description follows the workflow on Fig. 2.

Feature extraction

Corner interest points’ location Although many images in the EM dataset have low contrast and transparency challenges, the boundary and corner physical characteristics can easily be detected compared to other interest points. Therefore, we choose to use corner interest points. A corner is defined as the location (point) in the image where a slight shift in the location will lead to a large change in intensity in both horizontal (X) and vertical (Y) axes. It can also be defined as the intersection of points on objects contour edges, which retain important feature information of the objects (Peng et al. 2016). Harris and Shi and Tomasi corners theorems are among the most superior corner theorems (Harris et al. 1988) and (Shi and et al 1994). Simply these theorems operate on three steps:

Firstly, it is to determine the window, which produces high variation in intensity with a small move in the X and Y-axis. Numerically, to determine a window that can produce large variation, let the window be centered at (x, y) and the intensity at this point be I(x, y). I(x, y) is an individual intensity at a position, which can have a value from 0 to 255 for gray level image. When the window is shifted by (u, v), the intensity at the new location will be I(x + u, y + v) and [I(x + u, y + v) − I(x, y)] is the difference in intensity due to shift. For a corner, this difference must be high. Therefore, we maximize this term by differentiating it with respect to x and y. Let w(x, y) be the weights of pixels over the rectangular or a Gaussian window, Then, E(u, v) which is the difference between the original and the shifted window, is defined as:

$$E(u, v) = \sum_{x,y} w(x, y)[I(x + u, y + v) - I(x, y)]^2. \tag{1}$$

Apply the Taylor series with only the first order, which is$$T(x, y) = f(u, v) + (x - u)f_x(u, v) + (y - v)f_y(u, v). \tag{2}$$

Rewrite the shifted intensity using the above formula:

$$I(x + u, y + v) = I(x, y) + \frac{d(x, y)}{dx} (u) + \frac{d(x, y)}{dy} (v). \tag{3}$$

Let \( d(x, y)\) = \( I_x \) and \( d(x, y)\) = \( I_y \), then \( I_x \) and \( I_y \) are image derivatives in X and Y directions, respectively. So,

$$E(u, v) = \sum_{x,y} w(x, y)[I_x(u) + I_y(v)]^2.$$

Furthermore,

$$E(u, v) = \sum_{x,y} w(x, y)[I_x(u) + I_y(v)]^2. \tag{4}$$
Expand the above equation as:

\[ E(u, v) = \sum_{x,y} w(x, y)[I_x^2u^2 + I_y^2v^2 + 2I_xI_yuv]. \]  

(5)

where,

\[ M = w(x, y) \left( \frac{\sum I_x^2}{\sum I_y} \frac{\sum I_x I_y}{\sum I_y^2} \right). \]

(6)

where \( M \) is a symmetric 2 x 2 matrix whose eigenvalues are used to determine whether the scanned window contains a corner.

![Fig. 2](image.png)

Fig. 2 The workflow of the proposed method. (a) Is the stage for EM data preparation, (b) feature extraction, (c) feature pairing, (d) is for feature combination where pairwise features are combined with non-paired features, (e) classification of EMs using the combined features. The classifiers used are support vector machine with linear kernel (SVM-Linear), support vector machine with radial basis function kernel (SVM-Rbf), Logistic regression (LR), Random forest (RF), Linear discriminant analysis (LDA) and XGBoost.
Secondly, calculating the score value $S$ associated with the scanned window. For Shi and Tomasi corner (Shi and et al 1994), the score is given by:

$$S = \min(\lambda_1, \lambda_2),$$

(7)

And for Harris corner (Harris et al. 1988) the score is given by:

$$S = \det(M) - k(\text{trace}(M))^2$$

(8)

where $\lambda_1$ and $\lambda_2$ are eigenvalues of the matrix $M$, $\det(M) = \lambda_1 \lambda_2$, $\text{trace}(M) = \lambda_1 + \lambda_2$.

The third step is finding the points along the shift of the window, which can be regarded as corners. For the point to be regarded as the Shi corner, the score value $S$ should be greater than the specified value (if both the $\lambda_1$ and $\lambda_2$ are greater than the minimum threshold values).

In Harris theorem, the region is considered a corner if the score is larger than zero, which happens when both the $\lambda_1$ and $\lambda_2$ are greater than the minimum threshold values and their difference is small.

In the pilot experiment both theorems are tested to find the optimal corners. Shi and Tomasi theorem shows better performance by having more true corners which are within the microorganism body compared to Harris’ theorem. Thus, we choose to use Shi and Tomas theorem for the rest of the experiments. Shi and Tomasi theorem show superiority because of its stability, invariant to scale changes, translation, and rotation (Shi and et al 1994). Moreover, comparing with Harris corner points which we applied in our previous work (Zou et al. 2016), Shi and Tomasi give better results and more useful interest points than Harris’. Examples of images with corner points indicated on them are shown in Fig. 3.

Observing Fig. 3, the interest points are capable of detecting corner points which are rich in distinct information about the object of interest (microorganism) and isolate it from the unwanted background. In the practical case, we limit the number of corner points between 10 to 15 (due to the deep CNN model's computational complexity during the extraction of features).

Convolution neural network When a convolution neural network is trained on images similar to those in ImageNet dataset to perform image classification, it learns features that vary in complexity depending on the layers depth. On the upper layers, features similar to Gabor filters which act like edge and contour detectors are learned. These features are not explicit to a particular dataset. The lower layers in the network learn more composite features (from the combination of features of the upper layers) that resemble in forms and shape and are more specific to the present used dataset (Statnikov and Aliferis 2007). The learned parameters can be transferred to the new model whose lower layers might be different and the upper layers inherit the pre-trained network. The transfer learning technique has proven to improve performance when there is scarce of dataset for training (Shin et al. 2016), similar to the EM dataset. Among the most useful models which have been pre-trained on large ImageNet dataset are Vgg-16, Vgg19, ResNet-50, Inception-V3 (Liang et al. 2021). However, in this study we mainly focus on Vgg-16 because it has proven successful in many works when fine-tuned on other datasets for classification (Shin et al. 2016). Thus, to overcome the scarcity of EMs dataset which cannot train well the Deep CNN from scratch, we fine-tune the pre-trained VGG-16 using the EMs and extract deep learning features.

VGG-16 is a very deep convolution neural network for image recognition, proposed by Simonyan and Zisserman (2014). It is upgraded from AlexNet by replacing large-sized kernel filters (11 and 5) with $3 \times 3$. It has achieved high accuracy in many image classification tasks. It contains 21 layers with only 16 weight layers, which include 13 convolution layers with very small receptive fields of $3 \times 3$ (which gives its capability to capture the pattern of tiny information fields), followed by max pooling layers of size $2 \times 2$ and stride 2, which decreases the spatial resolution of the feature maps. In the end, there are three fully

![Fig. 3](image1.png)  
**Fig. 3** Detected Shi and Tomasi corner points (in green) on EMs
connected layers, which combine all learned features from previous layers and generalize them for classification. ReLu activation function is applied to all hidden layers. Lastly is the classifier layer. To leverage the fully connected (FC) layers, we extract deep learning features on the last FC layer. The dimension of each extracted feature is about $1 \times 1000$ dimensions. Figure 4 shows the VGG-16 network layers and the point from which deep learning features are extracted.

For each image, 10 patches (which are centered at interest points' coordinates) are meshed out and from each patch deep learning features are extracted. Then these 10 features for each image are stored parallel to their corresponding interest points' coordinates. Figure 5 summarizes the process of deep learning feature extraction. To get the optimal size of the patches, we used visual inspection to find the patches size that has the following characteristics: (1) it must cover enough region that has spatial characteristics/influence of a specific interest point where the patch is centered, (2) it allows very small interference area between nearby patches. Among the tested sizes $10 \times 10$, $20 \times 20$, $30 \times 30$, $40 \times 40$, and $50 \times 50$, the optimal dimension is found to be $40 \times 40$ pixel size for each patch as indicated in Fig. 5 part (a) and (b).

All experiments are conducted using a workstation with Intel (R) Core (TM) i77700 CPU with a speed of 3.60Hz, RAM of 32GB and NVIDIA GeForceGTX 1080 8GB. For implementation of the networks, we use python 3.6, Keras framework and Tensorflow 1.7 as the backend.

**Feature pairing: Delaunay triangulation (DT) theorem and middle point of the straight line theorem**

To pair feature maps that have been extracted from the interest points’ coordinates, we use the Delaunay triangulation (DT) theorem. DT theorem is one of the most robust graphical theorems for the representation of data. It is the theorem which forms triangles (Delaunay triangles) by connecting each data (coordinates) to its nearest neighbor, such that the circumcircle associated with each triangle does not contain a point in its interior (Khan et al. 2016). Geometrically, Delaunay triangulation for a given set $A$ of discrete data in a plane is a triangulation (DT), such that no data in $A$ is inside the circumcircle of any triangle in DT($A$). Delaunay triangulation maximizes the minimum angle of all the angles of the triangles in the triangulation (Delaunay et al. 1934). It is very effective for presentation of scattered data as it concentrates all data inside the major circumcircle formed by the most outer triangle as shown in Fig. 6 (b). Due to strong presentation power, it is used in many image matching works (Dou and Li 2014; Flores et al. 2017). Moreover, it is tolerable to spatial displacement of data (image objects) because it keeps the same association of the nearest objects within the image, regarded that the distortion is uniform all over the image.

The Delaunay triangle edges are formed by connecting nearest neighbor data points. This means two points (vertices) which share the same edge (line) have close
related characteristics (features). Thus, the middle point of 
the edge contains features which are an average of the 
edge end point features. Although (from our experiments) 
few middle points might be out of the EM’s body, which 
will have non similar characteristics between the edge end 
points; these points are very few (less than 5% of all 
the middle points). More than 95% of the middle points 
are within the main body of the EM (foreground) and 
have intermediate characteristics between the corresponding 
edge end points, as it can be observed in Fig. 6 (c). 
Owing to this, we pair the features that correlate to the 
vertices sharing the same edge to get the features of the 
middle point of edges. In this way, we increase the number 
of potential distinctive features for the microorganism, as 
shown in Fig. 6 (c). The pairing of features is done using 
the geometric theorem of the middle point of the straight 
line, because the edges of the triangles are straight lines. 
This is done by averaging the two feature vectors (maps) 
corresponding to each edge end coordinates as described 
in Eqs. 9 and 10. The edge coordinates are the interest 
points’ coordinates with their corresponding feature vectors 
(1 × 1000 dimensions) extracted from patches.

Let the coordinate of the two end points of an edge be 
represented by \((X_1, Y_1)\) and \((X_2, Y_2)\). The corresponding 
feature maps of the patches centered at these two points be 
\(F_1\) and \(F_2\). The middle point coordinate \(X_m, Y_m\) is given by:

\[
X_m, Y_m = \frac{(X_1 + X_2)}{2}, \frac{(Y_1 + Y_2)}{2}
\]  

(9)

The pairwise feature map \(F_m\), which corresponds to 
middle point \(X_m, Y_m\), is given by:

\[
F_m = \frac{(F_1 + F_2)}{2}
\]

(10)

On average, 36 to 43 pairwise features \((F_m)\) are formed 
from 10 original features for each image.
Joint pairwise feature formation

At this stage, we join 10 features formed on the interest points’ coordinates \((F_1, F_2,...)\) and pairwise features \((F_{m...})\). The average amount of pairwise features for each image is between 36 and 43. Thus, we form the joint feature maps by appending these features vertically. The average joint feature maps sizes range from \(46 \times 1000\) to \(53 \times 1000\) for different images. Each joint feature map corresponds to one original image. Because the dominant features are pairwise features, we name the features as joint pairwise features (pairwise deep learning features). To justify the potential of the pairwise deep learning features, we use them for the classification of EMs.

Classifiers

Six classifiers (from sklearn library) are used in this study to justify the usefulness of the pairwise features for the classification of microorganisms. These classifiers are support vector machine with linear kernel (SVM-Linear), support vector machine with radial basis function kernel (SVM-rbf), logistic regression (LR), linear discriminant analysis (LDA), XGBoost, and ensemble-based random forest classifier. In the SVM-Linear, the regularization is set to 1, the iteration count is not limited, and the numerical tolerance used is 0.001 to optimize the classifier. In the SVM-rbf, the decision function is set to “one vs. one” (ovo). LR classifier is used with Broyden-Fletcher-Goldfarb-Shanno (bfgs) as an optimization algorithm and the optimal number of iterations is found to be 600. In LDA, a singular value decomposition (SVD) is used for optimization and the absolute threshold for a singular value is set to 0.0001. The XGBoost is used with its default hyperparameters. The random forest classifiers are trained with 200 trees and the maximum depth is set to 21.

To evaluate and compare the classification results quantitatively, we use accuracy (Acc), F1-score, recall, precision (Prec), and specificity (Spec). These are the most standard metrics for the evaluation of classification performance. Table 1, summarizes the definition of these metrics.

From Table 1, true positive (TP) is the number of accurately predicted positive samples, true negative (TN) is the number of correctly classified negative samples, the number of negative samples classified as positive is false positive (FP), and the number of positive instances predicted as negative is a false negative (FN).

### Results and discussion

#### Selection of the corner detection theorem

Harris, and Shi and Tomasi corner theorems were tested on random images from different classes EMs. The best theorem is the one which detects much corners within the microorganism body. The corners within the microorganism are considered as True corners while those outside the microorganism are considered as False corners. A maximum required number of corners is set first and each theorem is tested if it can detect all corners as True corners. The results are indicated in Table 2. The results show that Shi and Tomasi perform better by detecting all corners as True corners in class 5 (c5), c9, c10, c18, c19, c20 and c21, while Harris theorem has detected all True corners only in c9, c19 and c20. Similarly in all other classes c2, c4, c7, c12 and c13 Shi and Tomasi detect more True corners than Harris. Therefore, Shi and Tomasi is adopted in this study for corner descriptors detection.

#### Evaluation of the non-paired features from patches

Firstly we mesh each image into patches which are centered at Shi Tomasi interest points. The size of each patch is \(40 \times 40\) pixel sizes. Then each patch is reshaped to \(224 \times 224\), which is the input size for the VGG-16 network. On average, each microorganism image results in 10 patches. Finally, deep learning features (feature vector) are extracted from each patch. Each feature vector is having a dimension of \(1 \times 1000\) size. Therefore, for each image 10 deep learning features are extracted. During the pilot experiments, we tested two options of combining the feature maps extracted from the same image. The first option is by appending all features horizontally to form a high dimension feature vector of \(1 \times 10000\) which represents one image. The second option is to treat each feature independently. This is similar to appending the features vertically resulting into \(10 \times 1000\) dimension matrix. In the later option, each feature from the same image points to the same label. In order to validate the best option which gives better performance, these features are used for binary image classification of class 1 and 2.

### Table 1  Definitions of used metrics

| Metric | Definition |
|--------|------------|
| Acc, Spec | \(\frac{TP + TN}{TP + FP + TN + FN}\) | \(\frac{TN}{TP + TN}\) |
| Recall | \(\frac{TP}{TP + FN}\) |
| F1-score | \(\frac{2TP}{2TP + FP + FN}\) |
| Prec | \(\frac{TP}{TP + FP}\) |
Table 2  Corner detection on different EM classes sample images. TAoC indicates the total amount of corners set, HTC is the amount of Harris’ True detected corners , and STTC is the Shi and Tomasi True corners detected.

| EM class | TAoC | HTC | STTC | EM class | TAoC | HTC | STTC |
|----------|------|-----|------|----------|------|-----|------|
| c2       | 36   | 16  | 15   | c12      | 35   | 34  | 26   |
| 31       | 31   | 14  | 13   | 31       | 14   | 13  | 12   |
| 43       | 43   | 22  | 20   | 43       | 22   | 20  | 16   |
| c4       | 26   | 25  | 25   | c13      | 39   | 37  | 34   |
| 22       | 22   | 22  |      | 22       | 22   | 22  |      |
| 24       | 24   | 24  |      | 24       | 54   | 51  | 47   |
| c5       | 10   | 10  | 8    | c18      | 85   | 85  | 84   |
| 12       | 12   | 12  | 11   | 12       | 91   | 91  | 90   |
| 13       | 13   | 13  | 11   | 13       | 92   | 92  | 91   |
| c7       | 26   | 23  | 4    | c19      | 14   | 14  | 14   |
| 31       | 31   | 25  | 4    | 31       | 19   | 19  | 19   |
| 37       | 37   | 30  | 4    | 37       | 16   | 16  | 16   |
| c9       | 28   | 28  | 28   | c20      | 17   | 17  | 17   |
| 31       | 31   | 31  | 31   | 31       | 20   | 20  | 20   |
| 24       | 24   | 24  |      | 24       | 15   | 15  | 15   |
| c10      | 21   | 21  | 19   | c21      | 14   | 14  | 13   |
| 26       | 26   | 26  | 23   | 26       | 22   | 22  | 21   |
| 30       | 30   | 30  | 28   | 30       | 33   | 33  | 32   |

of EMs. The Linear SVM is used for classification. The accuracy results obtained are 55.00% and 77.50% for the horizontal pending style and vertical style, respectively. Thus, we opt to use the vertical appending when performing the classification of 21 classes.

During image classification of 21 EM classes the features are randomly shuffled and partitioned at the ratio of 1:1 for training and testing sets. Then, classification of the EMs is performed into 21 classes. In order to reduce the bias of feature selection during shuffling, the whole process of shuffling randomly, partitioning, training, and testing is repeated 50 times. Each time the classification results (confusion matrices) are recorded. The average scores for all 50 iterations for each classifier are presented in Fig. 7.

The results in Fig. 7 show that all classifiers have achieved high accuracy and specificity while the F1-score, precision, and recall are still very low. This implies that the extracted features from interest points are potential features. However, the number of features representing the positive class (TP) during classification is less than the rest (TN) classes. To address this, we increase the number of potential features by pairing them, which will reduce the asymmetric of true class (TP) against the rest (TN) during classification.

**Evaluation of pairwise deep learning features**

In order to increase the number of potential features, the pairing of features is performed. After pairing, each image

![Fig. 7](https://example.com/fig7.png)  The average classification scores of 21 EM classes when using the non-paired deep learning features extracted from interest points (patches) only. The results are presented in percentage. RF denotes random forest, LDA is linear discriminant analysis and LR is Linear regression.
is represented by an average of 46 features resulted from pairing. These features are vertically appended. All features within the same image point to the same label. Then classification of 21 classes is performed. Similarly, in order to reduce the bias of some features on the classification results, cross validation is applied, where data are randomly shuffled then divided into the ratio of 1:1 for training and testing. The process from shuffling to classification is repeated 50 times for each classifier. The average results for all iterations are shown in Fig. 8.

Analyzing the results in Fig. 8, the pairwise features have significantly improved the classification results. Comparing with the average results of non-paired deep learning features in Fig. 7, the improvement of about 6.08%, 64.23%, 63.93%, 63.89%, and 3.20% in accuracy, F1-score, recall, precision, and specificity, respectively, is observed on SVM-Linear classifier. 2.13%, 22.84%, 22.22%, 23.50%, and 1.12% in accuracy, F1-score, recall, precision, and specificity, respectively, are observed on SVM-Rbf classifier. 4.69%, 49.93%, 49.03%, 49.26%, and 2.46% in accuracy, F1-score, recall, precision, and specificity, respectively, are observed on Random Forest classifier. 5.70%, 59.40%, 59.66%, 58.54%, and 2.98% in accuracy, F1-score, recall, precision, and specificity, respectively, are observed on Linear discriminant analysis (LDA) classifier. 4.95%, 51.98%, 51.71%, 51.90%, and 2.59% in accuracy, F1-score, recall, precision, and specificity, respectively, are observed on Logistic regression (LR) classifier. 4.83%, 51.30%, 50.69%, 51.14%, and 2.54% in accuracy, F1-score, recall, precision, and specificity, respectively, are observed on XGBoost. The improvement in F1-score, recall and precision shows that the pairwise features have better representation of the particular microorganism distinct features that can discriminate one microorganism class (TP) against others.

**Comparison of pairwise and non-pairwise deep learning features**

To validate the performance of the PDLFs more and ensure that the average results in “Evaluation of the non-paired features from patches” and “Evaluation of pairwise deep learning features” are not affected by outliers, we compare them against the maximum values of the classification results for both non-paired and pairwise features as indicated in Fig. 9.

Firstly observing the results in Fig. 9 and Figs. 7 and 8, the highest difference between the average and maximum results on pairwise feature-related graphs is 1.75% F1-score observed on the SVM-rbf while most of differences in other classifiers are less than 1.00%. The highest difference on non-paired feature results is 2.66% F1-score observed on SVM-Linear. These small difference values imply that the average results are not influenced by any outlier values.

Secondly, we compare the maximum results of the non-paired and paired features as observed in Fig. 9 only. A significant improvement of about 5.95%, 62.40%, 62.37%, 61.84%, and 3.23% in accuracy, F1-score, recall, precision, and specificity respectively, is observed on SVM-Linear classifier. 2.14%, 22.86%, 22.36%, 23.58%, and 1.11% in accuracy, F1-score, recall, precision and specificity respectively, are observed on SVM-rbf classifier. 4.57%, 48.44%, 48.05%, 47.79%, and 2.40% in accuracy, F1-score, recall, precision and specificity respectively, are observed on Random Forest classifier. 5.59%, 58.08%, 58.54%, 56.84%, and 2.93% in accuracy, F1-score, recall, precision and specificity respectively, are observed on LDA classifier. 5.17%, 54.42%, 54.26%, 54.19%, and 2.71% in accuracy, F1-score, recall, precision and specificity respectively, are observed on LR classifier. 4.82%, 51.12%, 50.79%, 51.27%, and 2.53% in accuracy, F1-score, recall, precision and specificity respectively, are observed on XGBoost classifier. The highest performing classifiers are SVM-Linear and LDA.

These results show that the linear classifiers (SVM-Linear and LDA) perform better than all other classifiers. The near coincident (close) results of SVM-Linear and LDA indicate that the pairwise features between classes have low correlation (Gokcen and Peng 2002). This implies that the pairwise features embody distinct discriminant characteristics to represent the microorganism. On the other
The maximum classification scores of 21 classes when using the non-pairwise deep learning features (Non-PDLF) extracted from interest points (patches) only which are represented with purple bars and the pairwise deep learning features (PDLFs) which are represented in green. All the results are presented in percentage.

|                      | SVM-Linear | SVM-rbf | Random Forest | LDA | Logistic Regression | XGBoost |
|----------------------|------------|---------|---------------|-----|---------------------|---------|
| Acc                  | 99.17      | 99.47   | 97.94         | 98.46| 97.89               | 98.99   |
| Prec                 | 93.22      | 93.33   | 93.37         | 93.26| 93.28               | 93.46   |
| Rec                  | 91.34      | 52.42   | 79.88         | 84.03| 77.85               | 93.26   |
| Spec                 | 91.48      | 96.38   | 96.21         | 94.81| 77.91               | 96.24   |
| F1                   | 91.32      | 52.42   | 79.88         | 84.03| 77.85               | 93.26   |

Fig. 9 The maximum classification scores of 21 classes when using the non-pairwise deep learning features (Non-PDLF) extracted from interest points (patches) only which are represented with purple bars and the pairwise deep learning features (PDLFs) which are represented in green. All the results are presented in percentage.

|                      | SVM-Linear | SVM-rbf | Random Forest | LDA | Logistic Regression | XGBoost |
|----------------------|------------|---------|---------------|-----|---------------------|---------|
| Acc                  | 99.17      | 99.47   | 97.94         | 98.46| 97.89               | 98.99   |
| Prec                 | 93.22      | 93.33   | 93.37         | 93.26| 93.28               | 93.46   |
| Rec                  | 91.34      | 52.42   | 79.88         | 84.03| 77.85               | 93.26   |
| Spec                 | 91.48      | 96.38   | 96.21         | 94.81| 77.91               | 96.24   |
| F1                   | 91.32      | 52.42   | 79.88         | 84.03| 77.85               | 93.26   |

Training and testing time evaluation

The estimated time for feature extraction is found to be 1.34 min for each image while feature pairing is about 0.04 s. The average training and testing time for each classifier when using the non-paired features and paired features for 210 images and one image are shown in Table 3.

From Table 3, the maximum time requirement for training is attained by XGBoost classifier, 1497.73 s (about 24.57 min) and 236.24 s (3.56 min) on PLDFs and Non-PDLFs respectively. This is because the XGBoost has an inherent feebleness in multi-class tasks when the number of classes is high. As the total number of training trees is the product of classes number and iteration. Therefore, the dimension of the PDLFs does not have any influence to XGBoost’s long time training. Considering only one EM image identification for the whole process from feature extraction, paring and testing, the total time for SVM-rbf on PDLFs is 80.89 s (1.35 min) while other classifiers are lower than that. Although the image testing time for the pairwise features is higher than the non-paired features for most of classifiers, they are generally very low (less than 1.35 min) and feasible for practical applications of classification tasks.

Comparison with other state-of-art techniques

We conduct some comparative experiments on classification of EMs using the well-known performing deep convolution (Vgg-16, Vgg-19, Inception-V3, and ResNet50) and hand-crafted feature-based techniques, gray level co-occurrence matrix (GLCM) and histogram of oriented gradients (HoG). The deep convolution neural networks (DCNN) are used for feature extraction and classification. For DCNN we adopt some experimental settings as in Liang et al. (2021), where the ImageNet pre-trained DCNN are used to serve the scarcity of EMs datasets. However, the augmentation
Table 3 Rows 1–4 are the training and testing time (in seconds) for 210 images when using non-paired deep learning features (Non-PDLF) from interest points only and when using pairwise deep learning features (PDLFs). The last two rows indicate the total time for feature extraction (FE), paring (PR) and testing (TE) on one image. RF represents Random forest, LDA is Linear discriminant analysis, LR is logistic regression and XGB is the XGBoost classifier.

|                     | SVM-Linear | SVM-Rbf | RF   | LDA | LR   | XGB   |
|---------------------|------------|---------|------|-----|------|-------|
| Train only, 210 images | Non-PDLF  | 8.02    | 7.81 | 13.04 | 2.3  | 7.73  | 236.24 |
|                     | PDLF      | 135.15  | 122.58 | 62.31 | 4.1  | 29.53 | 1497.73 |
| Testing only, 210 images | Non-PDLF  | 5.25    | 5.69 | 0.50 | 1.50 | 1.00 | 1.61 |
|                     | PDLF      | 74.97   | 94.64 | 2.50 | 0.92 | 1.41 | 1.03 |
| One image FE, PR,TE | Non-PDLF  | 80.47   | 80.47 | 80.44 | 80.44 | 80.45 | 80.45 |
|                     | PDLF      | 80.80   | 80.89 | 80.45 | 80.44 | 80.45 | 80.44 |

is not adopted in this study. Therefore, the small dataset is distributed at the ratio of 294:126 for training and testing.

In all the DCNN models we keep the upper layers of the original models and replace the fully connected layers with the following settings that performed best; For Vgg-16 and Vgg-19; after the flatten layer we add dropout layer of 10%, and then a dense layer of 2048 nodes, dropout layer of 30% and the classification layer of 21 nodes. In the ResNet-50, a global max pooling layer is added, followed by the dense layer of 1024 nodes, then the dropout layer of 50%, followed by dense layer of 1024 nodes, dropout layer of 30% and lastly the dense classification layer of 21 classes. For Inception-V3, after the last convolution layer, a global max pooling is added to flatten the feature maps followed by the dense layer of 512 nodes, then the dropout of 50%, and finally the classification layer of 21 classes. In all dense layers we use Relu as the activation function and Softmax is used in all classification layers. During experiment, the upper layers above flattening/global max pooling are frozen so they inherit the ImageNet pre-trained weights and the lower layers are fine-tuned using the EMs. The comparison results are shown in Fig. 10.

Moreover, we compare the PDLFs against Histogram of Oriented Gradients (HoG) and gray level co-occurrence matrix (GLCM). Similar dataset as in DCNNs is used. The features extracted from GLCM are energy, correlation, dissimilarity, homogeneity, and contrast. Same types of features are extracted at different variations of pixel distances 1, 3 and 5, while keeping the angle constant 0, then varying angles at $(\pi/4)$ and $\pi/2$ while keeping the distance at 0. This gives a total of 25 GLCM features for each image. For HoG features we abide by experimental settings recommended in Dalal and Triggs (2005), where the suitable image size used is $128 \times 256$ (ratio 1:2) pixel size for each image, the best orientation is found to be 9 pixel, cell size is $8 \times 8$ pixels, block is $4 \times 4$ and normalization is L2. Similar classifiers as in PDLFs are used to classify HoG and GLCM features. We only compare the results in accuracy and recall as they have shown the highest performance values in both HoG and GLCM. The comparison results between the PDLFs, HoG and GLCM features are shown in Fig. 11.

Observing the direct impact of PDLF when compared with the base model (Vgg-16) in Fig. 10, a major improvement of 22.15% F1-score, 21.07% recall, and 17.56% precision is achieved. This implies that the pairing of deep learning feature results in more potential features that highlight more the microorganism; thus, the network focuses its attention to the foreground. Similarly, the PDLF on SVM-Linear, Random forest, Linear discriminant analysis, Logistic regression and XGBoost perform better than all other DCCN which indicates the representation invariant and stability of the pairwise features that is they can constantly perform well in different classifiers. They inherit the stability of their base interest points in representing the discriminant regions of an object (microorganism) in an image.

Analyzing Fig. 11, both PDLFs and handcrafted features (HoG and GLCM) have shown higher performance in accuracy. Using accuracy only can not provide reliable conclusion because the results may be overruled by the true negative values only (Guo et al. 2008). Thus, to get more confidence we provide results in recall too. Comparing with HoG features, the PDLFs outperform by an increased recall of 48.11%, 8.53%, 34.54%, 46.04%, 30.04%, and 47.14% in SVM-Linear, SVM-rbf, Random forest, Linear discriminant analysis, Logistic regression and XGBoost respectively. And an increase of 69.90%, 34.50%, 47.82%, 59.67%, 59.14%, 47.48% compared with GLCM features when classified using SVM-Linear, SVM-rbf, Random forest, Linear discriminant analysis, Logistic regression and XGBoost respectively. A significant improvement observed by the PDLF is because they combine the strength of both deep learning and local features (handcrafted features).
Fig. 10  Comparison of the PDLFs when applied on SVM-Linear, SVM-rbf, Random Forest (RF), Linear discriminant analysis (LDA), Logistic regression (LR) and XGBoost classifiers, against DCNN which are Vgg-16, ResNet-50, Inception-V3 and Vgg-19. The DCNN are used for feature extraction and classification.

Fig. 11  Comparison between PDLFs, HoG features and GLCM when applied on Linear support vector machine (SVM-Linear), SVM-rbf, Random Forest (RF), Linear discriminant analysis (LDA), Logistic regression (LG) and XGBoost.
Moreover, the PDLFs are more specific to the features of the microorganism (as features are extracted from patches that are concentrated within the microorganism) than the GLCM and HoG which include some false features from the background.

Generally, the improvement of expressed by PDLFs in Figs. 8, 9, 10 and 11 indicates that the combination of deep learning and handcrafted features results in features which have better representation than the two.

Additionally, we compare this study with previous works for classification of EMs. In Li et al. (2013), the classification is based on four handcrafted features: shape features, edge histograms descriptor (including different categories of edges that are vertical, horizontal, 45 diagonal,135 antidiagonal and isotropic (non-edges)), Fourier descriptors, Geometrical features (which include perimeter, area, radii or extrema of axial lengths, seven invariant moments), and Internal structure histograms. This leads to a combination of 20 different feature spaces for each image. These features were applied for classification of only 10 classes of EMs using SVM which achieves an accuracy of 89.7%. In Yang et al. (2014), new nine geometric features were introduced for classification of 10 classes of EMs. The features are generated by considering the dimensions of equal length rectangular bounding boxes at three different widths sizes around the EM. Lastly SVM is used as a classifier which achieves an accuracy of 92.5%. Liang et al. (2021) used generic algorithm (GA) to optimize the fully connected layers of Inception-V3 for classification of 21 classes of EMs. In this work the GA is applied to fine tune the optimal values of the dropout rate and number of neurons for the fully connected layers of the inception-V3 so as to improve the classification rate and prevent over fitting. This result in the classification accuracy of 92.9%.

The PDLFs outperform the previous works due to high feature representation of the deep learning and pairing mechanism. Moreover, the use of 21 classes in PDLF justifies more the potential of the pairwise features having high performance than the use of only 10 classes in Li et al. (2013) and Yang et al. (2014).

Discussion

Each image is divided into 46 features during classification. Although the bulk of features are taken from within the microorganism body, some are extracted from outside the EM body (these are referred to as False features). False features might appear during the feature extraction (corner detection) phase, the pairing phase, or both. Due to poor contrast (between the EM body and background), non-uniformity or noise on some image backgrounds, points with high variation in intensity with a small shift in the X- and Y-axis (corners) can be detected as shown in Fig. 13 (a) by yellow colored points outside the microorganism body. During feature extraction the features centered at these false corners will have spatial feature values that do not correlate with other features pointing to similar labels (as other true features are extracted within the microorganism body). Moreover, during feature pairing the false corners (features) may result in false pairwise features which are outside the EM body as shown by green points on the second row of Fig. 13 (b). Additionally, the false features can be due to pairing of two True non-pairwise features whose edges center is outside the microorganism as shown by the green points on Fig. 13 (b) at the top row. All the False features contribute to the misclassification.

In general, the False features are low in number compared to True features in majority of the classes. In this aspect the misclassification of about 0.83% accuracy on SVM-Linear classifier is contributed by individual False features.

To analyze more the performance of pairwise features on each class, we present the average confusion matrices for each classifier in Fig. 12.

Observing the matrices in Fig. 12, the PDLFs have shown remarkable classification performance in almost all classes on all classifiers. This is because of the high capability of the pairwise features in capturing the distinct spatial characteristics of the microorganism in images. So, the pairing acts like an attention mechanism which reinforce the classification network to focus more on the microorganism features rather than the background features. However, in most classifier results (SVM-rbf, LDA and Random Forest), the most prominent least performing class is class 10 vorticella (c10) followed by class 5 Colpoda (c5). This is because our method has experienced challenges in extracting enough useful (true) features on classes 10 and 5. The biggest challenge on class 10 is the lack of distinct corner points on the microorganism (extremely low contrast between the microorganisms and background) as indicated on the second row of Fig. 13. Besides, the main challenge in class 5 is the noisy background as shown on the bottom row of Fig. 13. These have resulted in false corners, which lead to false pairwise features, and into misclassification experienced by almost all classifiers except SVM-Linear. Generally, the application of the best corner detection technique influences the results/potential of the pairwise features.
Fig. 12 Average confusion matrices showing the classification performance of each class on (a) Linear SVM, (b) SVM with rbf kernel, (c) Random Forest, (d) Linear Discriminant Analysis, (e) Logistic Regression, (f) XGBoost when using pairwise deep learning features.
Conclusion and future work

In this paper, the novel pairwise deep learning features are proposed. To increase the potential of deep learning features, a combination of handcrafted and deep learning is applied. Due to the high feature representation capability of interest points, we use Shi and Tomasi to represent distinct points on images where deep learning features are extracted. Then pairing is done between the nearby features based on interest points locations. The pairwise features prove their potential by obtaining the highest classification results on EMs images in all used classifiers. This technique is not only suitable for identification of EM images but also in medical-related microorganism images such Plasmodium from blood sample images and Tuberculosis bacteria from stained sputum samples. It can also find use in facial recognition tasks. Moreover, the patch-based deep learning features (without pairing) can be suitable in the classification of tumors such as brain and breast tumors from MRI or Ultrasound images when features are extracted from blob interest points locations. In future works, we will extract and pair the deep learning features from other prominent models such as ResNet50 and InceptionV3 and use them to perform the classification and segmentation of microorganisms.

Availability of data and materials The datasets analyzed during this study are available in the NEUZihan/EMDS-5 repository, link: https://github.com/NEUZihan/EMDS-5
Acknowledgements We thank Prof. Beihai Zhou, Dr. Fangshu Ma from the University of Science and Technology Beijing, PR China, and Prof. Yanling Zou from Freiburg University, Germany, for their previous cooperation in this work. We thank Miss Zixian Li and Mr. Guoxian Li for their important discussion. We also thank B.E. Xuemin Zhu from Johns Hopkins University, USA, and B.E. Bolin Lu from Huazhong University of Science and Technology, PR China, for their careful work in the EMDS-5 image data preparation.

Author contribution Frank Kulwa: conceptualization, investigation, methodology, software, writing original draft; Chen Li: conceptualization, investigation, methodology, software, supervision, resources, writing original draft, proofreading; Jinhua Zhang: methodology, proofreading; Kiniaki Shirahama: methodology, validation, resources; Sergey Kosov: methodology, validation; Tao Jiang: methodology, validation, proofreading; Marcin Grzegorzek: methodology, validation, resources; Xin Zhao: proofreading.

Funding This work received financial support from the “Natural Science Foundation of China” (No. 61806047).

Declarations

Conflict of interest The authors declare that they have no competing interests.

References

Afzali M, Ghaffari A, Fatemizadeh E, Soltanian-Zadeh H (2016) Medical image registration using sparse coding of image patches. Comput Biol Med 73:56–70
Bengio Y, Courville A, Vincent P (2013) Representation learning: a review and new perspectives. IEEE Trans Pattern Anal Machine Intell 35(8):1798–1828
Bolovinou A, Pratikakis I, Perantonis S (2013) Bag of spatio-visual words for context inference in scene classification. Pattern Recogn 46(3):1039–1053
Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE Computer society conference on computer vision and pattern recognition (CVPR’05), vol 1, pp 886–893
Delaunay B, Vide S, Lamémoire A, De Georges V (1934) Bulletin de l’académie des sciences de l’urss. Classe des sciences mathématiques et de nature (6):793–800
Dou J, Li J (2014) Image matching based local delaunay triangulation and affine invariant geometric constraint. Optik 125(1):526–531
Flores M, Torres G, García G, Licona M (2017) Fingerprint verification methods using delaunay triangulations. Int Arab J Inf Technol 14(3):346–354
Gokcen I, Peng J (2002) Comparing linear discriminant analysis and support vector machines. In: International Conference on advances in information systems, pp 104–113
Guo X, Yin Y, Dong C, Yang G, Zhou G (2008) On the class imbalance problem. In: 2008 Fourth international conference on natural computation, vol 4, pp 192–201
Harriss C, Stephens M et al (1988) A combined corner and edge detector. In: Alvey vision conference, vol 15, pp 10–524
Hinchey MG, Sterritt R, Rouff C (2007) Swarms and swarm intelligence. Computer 40(4):111–113
Keçeli AS, Keçeli SU, Kaya A (2018) Classification of radiolarian fossil images with deep learning methods. In: 2018 26th signal processing and communications applications conference (SIU), pp 1–4
Khan ZN, Qureshi R, Ahmad J (2016) On feature based delaunay triangulation for palmprint recognition. arXiv:160201927
Kosov S, Shirahama K, Li C, Grzegorzek M (2018) Environmental microorganism classification using conditional random fields and deep convolutional neural networks. Pattern Recogn 77:248–261
Kruk M, Kozer A, Osowski S, Trzciński P, Paszt LS, Sumorok B, Borkowski B (2015) Computerized classification system for the identification of soil microorganisms. In: AIP conference proceedings, vol 1648, p 660018
Kuzovkin I, Vicente R, Petton M, Lachaux JP, Baciu M, Kahane P, Rheims S, Vidai JR, Ara J (2018) Activations of deep convolutional neural networks are aligned with gamma band activity of human visual cortex. Communications Biology 1(1):1–12
Lazebnik S, Schmid C, Ponce J (2005) A maximum entropy framework for part-based texture and object recognition. In: Tenth IEEE international conference on computer vision (ICCV’05), volume 1, vol 1, pp 832–838
Li C, Shirahama K, Grzegorzek M, Ma F, Zhou B (2013) Classification of environmental microorganisms in microscopic images using shape features and support vector machines. In: 2013 IEEE international conference on image processing, pp 2435–2439
Li C, Wang K, Xu N (2019) A survey for the applications of content-based microscopic image analysis in microorganism classification domains. Artif Intell Rev 51(4):577–646
Li C, Kulwa F, Zhang J, Li Z, Xu H, Zhao X (2020) A review of clustering methods in microorganism image analysis. Information Technology in Biomedicine, 13–25
Liang CM, Lai CC, Wang SH, Lin YH (2021) Environmental microorganism classification using optimized deep learning model. Environ Sci Pollut Res, 1–13
Lindeberg T (2013) Scale selection properties of generalized scale-space interest point detectors. Journal of Mathematical Imaging and Vision 46(2):177–210
Ling H, Soatto S (2007) Proximity distribution kernels for geometric context in category recognition. In: 2007 IEEE 11th international conference on computer vision, pp 1–8
Liu D, Hua G, Viola P, Chen T (2008) Integrated feature selection and higher-order spatial feature extraction for object categorization. In: 2008 IEEE conference on computer vision and pattern recognition, pp 1–8
Luo JY, Irisson JO, Graham B, Guigand C, Sarafraz A, Mader C, Cowen RK (2018) Automated plankton image analysis using convolutional neural networks. Limnol Oceanogr Methods 16(12):814–827
Mithra K, Emmanuel WS (2019) Automated identification of mycobacterium bacillus from sputum images for tuberculosis diagnosis. SI ViP 13(8):1585–1592
Morikoa N, Satoh S (2010) Building compact local pairwise codebook with joint feature space clustering. In: European conference on computer vision, pp 692–705
Morikoa N, Satoh S (2011) Compact correlation coding for visual object categorization. In: 2011 International conference on computer vision, pp 1639–1646
Peng W, Hongling X, Wenlin L, Wenlong S (2016) Harris scale invariant corner detection algorithm based on the significant region. International Journal of Signal Processing Image Processing and Pattern Recognition 9(3):413–420
Priya E, Srinivasan S (2016) Automated object and image level classification of tb images using support vector neural network classifier. Biocybernatsics and Biomedical Engineering 36(4):670–678
Rajaraman S, Candemir S, Kim I, Thoma G, Antani S (2018) Visualization and interpretation of convolutional neural network predictions in detecting pneumonia in pediatric chest radiographs. Appl Sci 8(10):1715

Savarese S, Winn J, Criminisi A (2006) Discriminative object class models of appearance and shape by correlatons. In: 2006 IEEE Computer society conference on computer vision and pattern recognition (CVPR’06), vol 2, pp 2033–2040

Schmid C, Mohr R, Bauckhage C (2000) Evaluation of interest point detectors. International Journal of computer vision 37(2):151–172

Shi J et al (1994) Good features to track. In: 1994 Proceedings of IEEE conference on computer vision and pattern recognition, pp 593–600

Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. IEEE Transactions on Medical Imaging 35(5):1285–1298

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:14091556

Singh RL. (2017) Principles and applications of environmental biotechnology for a sustainable future

Statnikov A, Aliferis CF (2007) Are random forests better than support vector machines for microarray-based cancer classification? In: AMIA Annual symposium proceedings, american medical informatics association, vol 2007, p 686

Statnikov A, Aliferis CF, Tsamardinos I, Hardin D, Levy S (2005) A comprehensive evaluation of multcategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics 21(5):631–643

Wang YH (2014) Research on segmentation of protozoan and metazoan image in microscopic examination of activated sludge.

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.