Omics Imagification: Transforming High-throughput Molecular Representation of a Cell into an Image

Derong Liu §, Seid Miad Zandavi §, Vera Chung, Ali Anaissi, Fatemeh Vafaee*
§ equal contribution; Co-first authorship
* Corresponding Author

***Abstract*— Different omics profiles, depending on the underlying technology, encompass measurements of several hundred to several thousands of molecules in a biological sample or a cell. This study develops upon the concept of "omics imagification" as a process of transforming a vector representing these numerical measurements into an image with a one-to-one relationship with the corresponding sample. The proposed imagification process transforms a high-dimensional vector of molecular measurements into a two-dimensional RGB image to enable holistic molecular representation of a biological sample and to improve the classification of different biological phenotypes using automated image recognition methods in computer vision. A transformed image represents 2D-coordinates of molecules in a neighbour embedded space representing molecular abundance and gene intensity. The proposed method was applied to single-cell RNA sequencing (scRNA-seq) data to "imagify" gene expression profiles of individual cells. Our results show that a simple convolutional neural network trained on these single-cell transcriptomics images accurately classifies diverse cell types outperforming the best-performing scRNA-seq classifiers such as Support Vector Machine and Random Forrest.**

***Index Terms*— Image Transformation, CNN, Single-cell RNA Sequence, Non-image Data**

I. Introduction

MODERN data-driven biology relies on the interpretation of large-scale molecular measurements (known as '*omics*' data) to understand and predict biological *phenotypes* such as the characteristics of an organism or the state of an individual cell. Next-generation sequencing (NGS) biotechnologies can now comprehensively measure different types of molecules (e.g., DNA, RNA, and proteins) at a high-speed and low-cost, leading to an unprecedented generation of *omics* data from different biological samples or across thousands of individual cells. The 'omics' notion indicates that nearly all instances of the targeted molecules are measured in the assay providing holistic views of the biological system [1]. For instance, *transcriptomics* data derived from RNA-sequencing (RNA-seq) technologies measure the quantity of over $20,000$ RNAs in a biological sample at a given moment.

Conventional RNA-seq experiments measure the *average* RNA expressions from large populations of cells within a sample or tissue section. Such average measurements, however, may obscure critical differences between individual cells within these populations [2]. In contrast, *single-cell* RNA sequencing (scRNA-seq) provides transcriptomics profiles of thousands of individual cells providing unprecedented opportunities to characterise the cellular composition of complex tissues. Analysis of single-cell sequencing data has taken the front seat in deriving useful insights into cellular functions and identification of cell sub-types with diverse applications across biological disciplines including developmental biology, neurology, oncology, immunology, and Infectious disease.

A common step in analysing single-cell data involves the identification of cell populations present in a given sample. This task is accomplished either via unsupervised clustering [3] (i.e., grouping cells based on similarities in their expression profiles followed by cell annotation, which involves manual inspection of cluster-specific marker genes) or via supervised classification enabling automated cell identification and annotation [4]. The latter circumvents the cumbersome and time-consuming step of manual cell annotation and has increasingly become popular due to the growing number of annotated scRNA-seq datasets available for training classifiers [5]. A recent study has benchmarked over 20 classification methods (including both single-cell-specific and general-purpose classifiers) to automatically assign cell identities across several scRNA-seq datasets of different sizes, technologies, and complexity and demonstrated that the general-purpose support vector machine (SVM) classifier is the best performer, overall [4].

However, SVM, similar to other traditional classifiers, requires a feature selection step prior to training to identify informative features (e.g., differentially-expressed genes) or latent variables (e.g., features extracted via dimensionality reduction) as predictors of the classifier. The choice of feature selection often significantly affects classification performance and further complicates the application of a classifier for cell

Derong Liu, Vera Chung and Ali Anaissi are with the School of Computer Science, The University of Sydney, Sydney, Australia (email: jimmyliu537@gmail.com, vera.chung@sydney.edu.au, ali.anaissi@sydney.edu.au).
Seid Miad Zandavi and Fatemeh Vafaee are with School of Biotechnology and Biomolecular Sciences, The University of New South Wales (UNSW), Sydney, Australia (email: s.zandavi@unsw.edu.au, f.vafaee@unsw.edu.au)

identification [6]. Additionally, while SVM is proven to be a powerful scRNA-seq classifier, it shows varied performance across different datasets and fails to correctly determine cells in complex populations which incorporate several closely-related cellular subtypes [7]. This signifies the intricacy of expressional patterns underlying different cell types and the need for an end-to-end learning platform that can decipher subtle differences among cell types.

Deep artificial neural networks (ANNs) are widely acknowledged for their ability to perform *automatic feature extraction* from raw data. Instead of relying on a pre-determined hand-crafted set of features, deep learning algorithms automatically learn features directly from data [8]. Hence, a deep learning-based model has the capacity to develop a single-step phenotype prediction procedure where omics data would be directly used to classify the outcome without prior feature selection. The absolute superiority of convolutional neural networks (CNNs) in image classification has been widely acknowledged but not yet sufficiently leveraged for classifying cell types based on single-cell *omics* data.

Convolutional Neural Network (CNN) has shown promising results in many areas. One downside of CNN is that it only accepts images as input. This study aims to convert data that are not in the form of images to image data that CNN can process. The priority is to develop an image data transformation pipeline suitable for single-cell data. With the maturity of machine learning theories and the rapid development of network speed and computational power, researchers are encouraged to experiment on a variety of machine learning classifiers such as K Nearest Neighbour, Decision Trees, Naive Bayes, Random Forest and Support Vector Machine. The main challenge is to compose a classification model on cell data. This also applies to other biological or genetic data because they usually contain a large number of features in each example. This high dimensional property is a significant suppression of the performance and efficiency of conventional machine learning frameworks. Moreover, traditional machine learning methods demand a one-dimensional feature vector through feature extraction. The resulting feature vector is the blueprint for the upcoming training process, data passed in the model are categorised based on the feature vector. For each example in the dataset, features in the feature vector are broadly viewed as independent of one another, meaning the change of one feature imposes mere or no effect on the remaining features, regardless of the order of features.

Despite CNN's impressive results on image form data, it still struggles to work with non-image data productively and effectively. Copious data are not in the form of images and are still widely used in other machine learning models, for instance, music note, mutation, message, RNA-seq and financial time series data. CNN accepts images as inputs (2D $m \times n$ matrices) and performs feature extraction and classification on them through its hidden layers. CNN is prevalent in today's commercial application due to its unique advantages over other models. For example, CNN can achieve good levels of performance from fewer training data and less training time. If we can find a way to convert non-image data to image data, we then have the opportunity to run CNN on these converted images since CNN only accept images as input. CNN has shown promising results in image data classification. It is exciting to see what CNN has to offer us on these non-image data. CNN takes in an image as input. A picture is made up of pixels, and adjacent pixels share similar information. However, if pixels are in arbitrary arrangements, this would make it difficult for CNN to find patterns in local regions. As a result, independent pixels with misarranged placement can negatively impact the performance of feature extraction and classification. Thus, the data transformation with consideration to spatially coherent pixels in local regions is crucial. Such transformation helps to improve the resolution of cell types in the single-cell data.

In this study, we presented a method to convert a vector representing scRNA-seq numerical measurements to an image (matrix data keeping spatial image information), referred to as 'omics imagification' for cell-type identification using CNN-based classification models. The proposed method is inspired by Fast Fourier Transform(FFT)[9], which is a fast algorithm for computing Discrete Fourier Transform(DFT)[10]. DFT is to explore the characteristics of complex waves such as amplitude, phase and frequency in form of complex numbers. The real and imaginary parts of these complex numbers are essentially two sets of real values that are required to represent the two-dimensional coordinates of the output as a function of frequency that has both phase and magnitude. Here, the idea of signal in the frequency domain is applied to single-cell image transformation as the resulting complex values contain most of the information of the original one-dimensional vector, and provide fundamental geometric structure for 2-dimensional image mapping.

II. Related Works

Previous studies have been conducted to use different CNN architectures on various datasets, trying to overcome the obstacle that CNN only accepts images as input. Lyu and Haque conducted bioinformatics researches on Tumor type classification based on deep learning [11]. The gene expression data they used are categorised as non-image data. They tried to convert gene expression data to a $2D$ image before applying deep learning models to investigate tumour type classification. They could accomplish such data transformation by first ordering genes based on chromosome number. The physical distance between genes is an essential factor that contributes to the interaction between genes. They reshaped the original data from a 1×10381 long vector to a 2D matrix of 102×102.

Next, Jian Yang and Jing-yu Yang proposed an image projection technique called image principal component analysis (IMPCA) to directly produce the image total covariance matrix from the original image matrix [12]. PCA is a vector-based analysis technique. For images recognition problem, a transformation is required from image matrices to dimensional vectors. PCA has its limitation. A high-resolution image in this technique corresponds to a high dimensional vector. Such high dimensional image vectors significantly increase PCA's processing time, making it less efficient when dealing with image recognition. IMPCA can offer better computational time

than PCA, where the image size of the covariance matrix produced by IMPCA is smaller than that of PCA.

Further, Kovalerchuk and his colleagues [13] proposed a new CPC-R algorithm to help convert non-image data to images through pair values mapping. It requires two times fewer cells compared to single value mapping and shows promising results. The general idea is to group numerical attributes in pairs (x_i, x_{i+1}), and such pair (x_i, x_{i+1}) is mapped to a 2D plane corresponding to the horizontal (x_i) and vertical (x_{i+1}) coordinates. Then they label cells with various intensities of grey-scale from black (x_1, x_2) to extreme light grey (x_{n-1}, x_n). Although they kept improving the existing pipeline and came up with updated versions of the CPC-R algorithm, their theory's core ideas are mentioned above in their base CPC-R algorithm.

Accordingly, researchers [14] transformed gene-expression data from vectors to images that CNN can exploit. They started by rearranging the position of the genes based on biological criteria, KEGG BRITE hierarchical information is used as a guideline to order the gene-expression data. Hence, they can construct the gene-expression images with the generated hierarchical structure and then use them for cancer survival prediction. Additionally, Anders proposed a way to visualise genomic data with the Hilbert curve [15]. The Hilbert curve is a continuous fractal space-filling curve. As an effective technique for linear mapping of multidimensional space, the linear space preserves the locality of features adequately in multidimensional space [16]. The superior clustering properties of the Hilbert space-filling curve suggest its potential in dealing with data mapping between dimensions.

To utilise the potential of CNN for non-image data of breast cancer as it reduces the necessity for laborious feature extraction, Sharma and Kumar proposed three methods of data wrangling that convert a 1-D vector to a 2-D image matrix [17]. The first method involves using the bar graph to visualise every feature of the dataset. The width of the graph is $xd + y(d + 1)$ where x is the width of a bar. d is total features and y is the gap between adjacent bars. The height is normalised to create a square image. The order of the bars depends on the similarity of features and is crucial for CNN to depict patterns. In essence, this is a shortest-path problem. Numerous meta-heuristic algorithms [18] such as Particle Swarm Optimisation [19], Reincarnation Algorithm [20] and Genetic Algorithm [21] can be used to retrieve the optimal order of bars. The second method is the use of the normalised Euclidean distance matrix. The key concept is the difference between two features i.e. $x_{ab} = x_a - x_b$ where x_a and x_b are the values of a given feature with (a, b) belong to $[1, d]$. A square image of size $[d \times d]$ is created. Compared to bar graphs, this graph is three times longer. The third method is a combination of methods 1 and 2. They created a three layers image of size $[3d \times 3d]$ where a copy of numerical data, the bar graphs and a normalised distance matrix stack on top of each other. It is evident that the third method outperformed two other methods because it perceives the most information about the data.

Finally, the DeepInsight method introduced by Sharma et al. [22] transforms non-image data into images by putting similar elements and features together and separate dissimilar ones further away from each other. This collaborative approach to element arrangement may aid in discovering hidden structures or the comprehension of the relationships between a group of features. They use t-SNE [23] to measure similarity while t-SNE have effects on the entire dataset. This means that the converted images are based on the whole cohort of observations rather than a single observation. If the order of observations and features in the entire dataset has changed, resulting in the change of converted images. Based on the DeepInsight theory, they introduced a practical pipeline for CNN on non-image data called DeepInsight-Feature Selection Algorithm (DeepInsight-FS) [24], which contains several crucial stages such as element selection, feature arrangement, feature derivation and classification. Results suggest that DeepInsight-FS has advantages over traditional classification algorithms at identifying significant gene groupings that were related to the phenotype of interest and its potential at interpreting sophisticated omics data.

In this paper, we propose a novel image transformation method to convert non-image Omics data into images for classification purposes, applied to single cell RNA-seq data. To explore various cell types, one-dimensional gene expression vectors are converted into two-dimensional images using Fast Fourier Transform (FFT). We can then utilise the power of CNN on $2D$ images classification problems to better distinguish local patterns among features in each sample.

III. Problem Formulation

Different omics footprints might include measurements of hundreds to thousands of elements, based on the underlying method employed. The general transformation process involves converting a vector containing these numerical data into an image with a one-to-one correspondence to the relevant sample. Basically, A 2D image is made up of spatially coherent pixels that exchange important information with neighbouring pixels. Pixel placement can have a detrimental influence on feature extraction performance and classification accuracy if they are organized randomly. Thus, it is important to retain the data integrity and prevent excessive loss of information during image transformation.

In this paper, we introduce a method to convert non-image data into images. Converted images can then be used in CNN for general image classification tasks. Firstly, we apply Log Normalisation on raw data. Then, for each sample, we use the Fast Fourier Transform(FFT) algorithm [25] to compute a vector consisting of real and imaginary values. Extract the real and imaginary parts of the vector as vector z_{real} and $z_{imaginary}$, respectively. Following this procedure, apply the mean values of z_{real} and $z_{imaginary}$ across all samples is applied in order to have $z_{mean} = (z^m_{real}, z^m_{imaginary})$ as reference frames in the Cartesian plane. Find the convex hull on the real as x-axis and imaginary as y-axis before rotating the convex hull by the rotation matrix. Convert the Cartesian rotated axis into the pixel frame coordination. Once we have the coordinate frame, we can fit the normalised data into corresponding coordinates. Figure 1 is a flowchart that illustrates an overview of the entire operation.

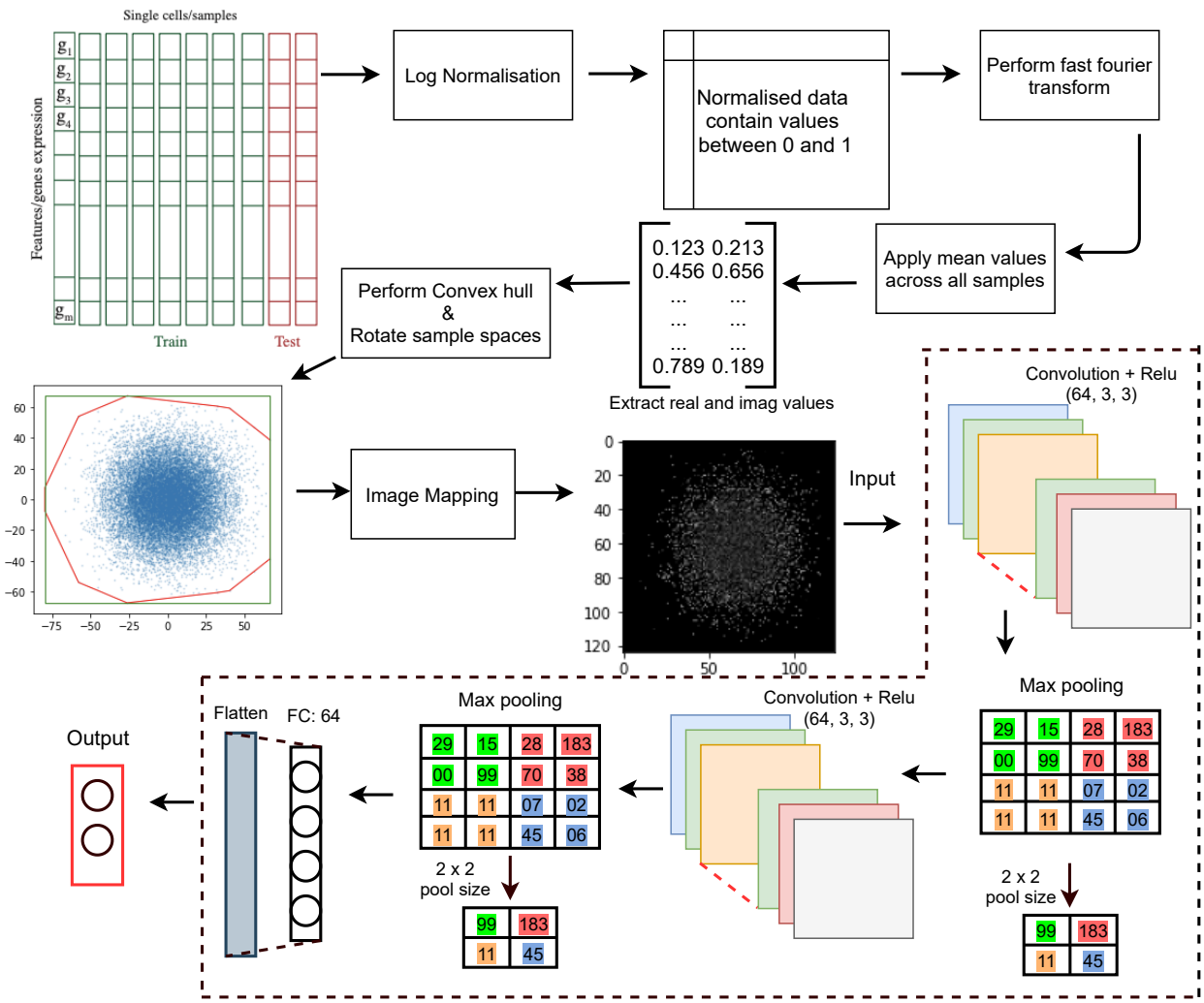

Fig. 1. FFT structure and CNN architecture flowchart

To initiate the image conversion process, a data prepossessing method is deployed to perform data normalisation. A normalised dataset generally reduces computation time and lead to faster convergence. Deep learning algorithms' effectiveness is dependent on the quality of the data used to generate a comprehensive prediction model of the classification task [26]. In this method, Log normalisation is the primary data prepossessing approach. It will adjust the feature values between 0 and 1. The maximum value is obtained through the logarithmic scale to set the feature values between 0 and 1. The minimum values are changed accordingly for each feature or attribute. The following equations (Eq.1, 2, 3, 4) describe how the maximum and minimum values can be computed.

$$X^i_{min} = \min X_{ds}(i,:) \tag{1}$$

$$X_{ds}(i,:) = \log(X_{ds}(i,:) + |X^i_{min}| + 1) \tag{2}$$

$$X_{Max} = \max(X_{ds}) \tag{3}$$

$$X_{ds}(i,:) = \frac{X_{ds}(i,:)}{Max} \tag{4}$$

Where X_{ds} is the training dataset and $(i,:)$ represents all the samples of the ith feature. Hence, X^i_{min} is the minimum value of the ith feature. The validation and test sets are computed using the training extrema values. Especially, any feature values larger than 1 after normalizing by the maximum value X_Max will be rounded down to 1. Likewise, after adjusting by the minimum values X^i_{min}, any normalized values of validation and testing set will be set to 0.

Next, we find the minimum convex that encompasses the feature matrix. To determine the smallest convex set, the convex hull is used. As CNN accepts images that are framed either vertically or horizontally, we need to perform rotation. See Eq.5.

$$\begin{bmatrix} X_{ri} \\ Y_{ri} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \end{bmatrix} \tag{5}$$

where $[X_i, Y_i]^T$ is the resulting images after normalization FFT and $[X_{ri}, Y_{ri}]^T$ is the rotated plane.

To find the features at the correct locations, the Cartesian plane must be transformed to spatially pixel forms. The conversion process from the Cartesian plane to pixel frames can be conducted utilizing the mean of the attributes in the

same position. See Eqs.6 and 7.

$$x_c = round\left(\frac{P_x \times (X_{ri} - \min(X_{ri}))}{\max(X_{ri}) - \min(X_{ri})} + 1\right) \quad (6)$$

$$y_c = round\left(\frac{P_y \times (Y_{ri} - \min(Y_{ri}))}{\max(Y_{ri}) - \min(Y_{ri})} + 1\right) \quad (7)$$

where x_c, y_c are computed pixel coordinates, P_x and P_y are pre-determined pixel size, and X_{ri}, Y_{ri} are rotated image planes.

The pixel intensity is determined by the mean values of the features in the same location. Therefore, when the picture is very tiny, plenty of the attributes or characteristics overlap, and the visual representation could be inaccurate when compared to the number of features available. Pixel size needs to be chosen carefully while considering the hardware capability and the composition of data sets. The algorithm 1 demonstrates the process of FFT Omics Image Transformation in pseudo-code.

Algorithm 1: FFT Omics Image Transformation

Data: Omics data long vector $V = \{v_1, v_2, ..., v_n\}$
Result: Transforming $1D$ numerical vector of Omics sequences to a $2D$ image $I = \{i_1, i_2, ..., i_n\}$.

```
begin
    Determine pixel size;
    Data normalisation using Log Scalar;
    begin
        Perform Fast Fourier Transform;
            Extract real and imaginary parts from
          resulted vector;
            [z_r, z_i] ⟵ resulted vectors;
            apply mean values of [z_r, z_i];
        generating convex hull;
        rotating convex hull plane;
    end
    for v ∈ V do
        - map features into pixel frame coordination;
        - compute matrix coordinates of each feature
          based on the pixel dimensions;
        - convert matrix coordinates to rgb channels;
        - scale transformed 2D images;
    end
end
```

IV. Data Collection

A total of 14 datasets is obtained from a curated publicly available scRNA-seq dataset maintained by the Hemberg Lab [27]. Each dataset includes labels for predicted cell types, which are used as the ground truth of classifier accuracy. These datasets are selected to include a wide range of protocols, tissue types, and dataset sizes. Datasets include different species (human and mouse), tissues (colorectal tumor, embryo, heart, colon, liver) and various protocols (SMARTer, Smart-Seq2, Chromium). In the identification of different cell types, transcriptomic reads and unique molecular identifiers (UMIs), counts can play an important role in compensating for variation in the sequencing depth when comparing these measurements across cells. Accordingly, two different read types (i.e., read counts and UMIs) are considered here. We obtained human colorectal tumor and mouse embryo datasets from the following accession numbers: GEO: GSE81861 (SMARTer), GEO: GSE45719 (SMART-Seq2), and TabulaMuris cells sequenced with SMART-Seq2 and Chromium, as listed in Table I.

V. Results and Discussion

The FFT omics imagification technique presented is a transformation method for converting a one-dimensional vector into 2-dimensional images. The technique arranges spatially coherent pixels in sample spaces by utilizing an algorithm (e.g. FFT) to compute discrete Fourier transform of a data sequence. In this paper, 14 available single-cell datasets were chosen to analyse the performance of machine learning and deep learning frameworks on transformed omics image data.

Accordingly, data normalization is a crucial pre-processing step for the omics data [7]. It can be to mitigate the negative impact of noise and bias in the transformation process, providing a relatively stable environment for the extraction of features or attributes. Several studies have indicates that a wide range of strategies for single-cell data normalization have been utilized such as high volume single-cell data and micro-array data [28]. Such techniques tend to overlook important characteristics of a sequence. For instance, zero inflation refers to the zero-read counts seen in numerous single-cell sequences [29]. Here, three forms of data normalization are studied, raw data, Count Per Million (CPM) [30] and Minimum Maximum scaling [31] on CPM data (MinMax(CPM)).

An example of image-formed data with its density of distinct cell types per pixel is illustrated in Figure.2. This method captures key information from one-dimensional vector data, extracting features, rearranging elements and projecting resulting matrix on a two-dimensional plane.

To assess the proposed method's performance, we utilised three traditional machine learning frameworks and the most recent non-image to image data transformation method, DeepInsight (DI) [24]. The related machine learning classifiers used for benchmarking are K-Nearest Neighbors (KNN) [32], Random Forest (RF) [33] and Support Vector Machine (SVM) [34]. Moreover, an identical CNN architecture (Figure 1) was used on both FFT and DeepInsight (DI) transformed images for classification. This study incorporates 14 various single-cell datasets to obtain a comprehensive performance evaluation. Each dataset is expected to experience a train-test data split procedure. Training set occupy 80% of all examples, testing set take up the remaining 20% and 20% of the training set is used as the validation set.

To offer an impartial evaluation of the model, the classification accuracy and macro F1-score were computed on the test set applying 5-fold cross-validation. The accuracy is described as the number of correct predictions as a percentage of the test set, where the F1-score can be considered as a weighted average of the recall and precision. To better visualize the

	Dataset	Species	Tissue Type	Protocol	Genes	Cells	Classes	Read Type
1	deng-reads	Mouse	Embryo	SMART-Seq2	22431	268	6	Read
2	TabulaMuris-Heart-10X	Mouse	Heart	Chromium	23433	654	6	UMI
3	zhengmix4uneq	Human	Immune	GemCode	16443	6498	4	UMI
4	zeisel	Mouse	Brain	STRT-Seq	19972	3005	9	UMI
5	yan	Human	Embryo	Tang	20214	90	6	Read
6	TabulaMuris-Liver-10X	Human	Liver	SMART-Seq2	23433	1924	3	Read
7	fan	Mouse	Embryo	SUPeR-seq	26357	66	6	Read
8	zhengmix4eq	Human	Immune	GemCode	15568	3994	4	UMI
9	darmanis	human	brain	SMARTer	22088	466	9	Read
10	TabulaMuris-Marrow-10X	human	Marrow	SMART-Seq2	23433	4112	9	Read
11	TabulaMuris-Trachea-FACS	human	Trachea	SMART-Seq2	23433	1391	5	Read
12	tasic-reads	Mouse	Brain	SMARTer	24150	1679	18	Read
13	TabulaMuris-Pancreas-FACS	Mouse	Pancreas	Smart-Seq2	23433	1961	10	Read
14	manno-mouse	Mouse	Brain	STRT-Seq	24378	2150	32	UMI

TABLE I
SINGLE CELL DATA USED FOR OMICS IMAGIFICATION

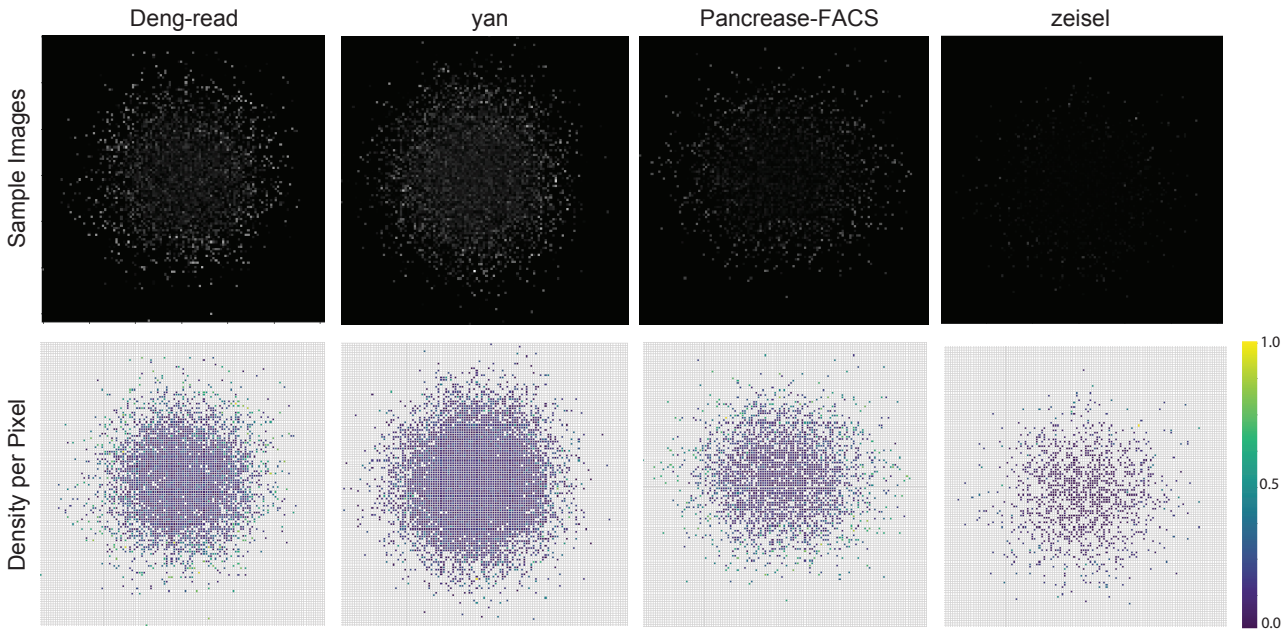

Fig. 2. Sample images of RNA-seq data and the density of RNA features per pixel

variance of each model, heat maps are utilised based on experimental results because of it's ability to communicate numerical values productively with colour. Figure 3, 4 and 5 reveal test accuracy and $F1 - scores$ of machine learning models (KNN, RF, SVM) on raw, CPM and MinMax normalized data, respectively. These figures also contain metrics of the CNN model on two different imagification methods (FFT and DeepInsight (DI) across 14 datasets.

The convolutional neural network(CNN) architecture described in Figure 1 includes a combination of a convolution layer, plus a ReLU activation function, then a max-pooling layer with a pool size of 2×2. There are two of these combination sequences, followed by a fully connected and a flattened layer. In this experiment, the input pixel size was predetermined to 124×124. The pixel size plays an important role in the performance of the model. If the pixel size is too small, only limited information is displayed on the converted image, negatively impacting the resulted accuracy. However, while choosing a larger pixel size could keep more information, it could slow down the model and make it less efficient. Overall, the FFT approach has demonstrated a competitive and consistent result across all 14 datasets with different data pre-processing methods. Three machine learning models(KNN, RF, SVM) suffers from poor performance in scRNA-seq data. For example, the RF model did not perform well in the Zeisel dataset with an F1 score of 22% for raw data, which was significantly lower than others, FFT(89%), KNN(85%) SVM((68%) and DI(22%). Similarly, the SVM model did poorly in the liver-10X dataset with an F1 score of 50% where the rest have an average score of 97%. The KNN model in the darmanis dataset had the lowest F1 score of 17%. On the other hand, the FFT model has shown a high level of consistency and outperformed its peers multiple times. Furthermore, in comparison to the popular DeepInsight method [22], the FFT model still performed strongly. In total of 14 datasets, DeepInsight had average F1 scores of 88%(raw), 78%(CPM) and 77%(MNX), while FFT returned (90%)(RAW), 80%(CPM) and 76%(MinMax).

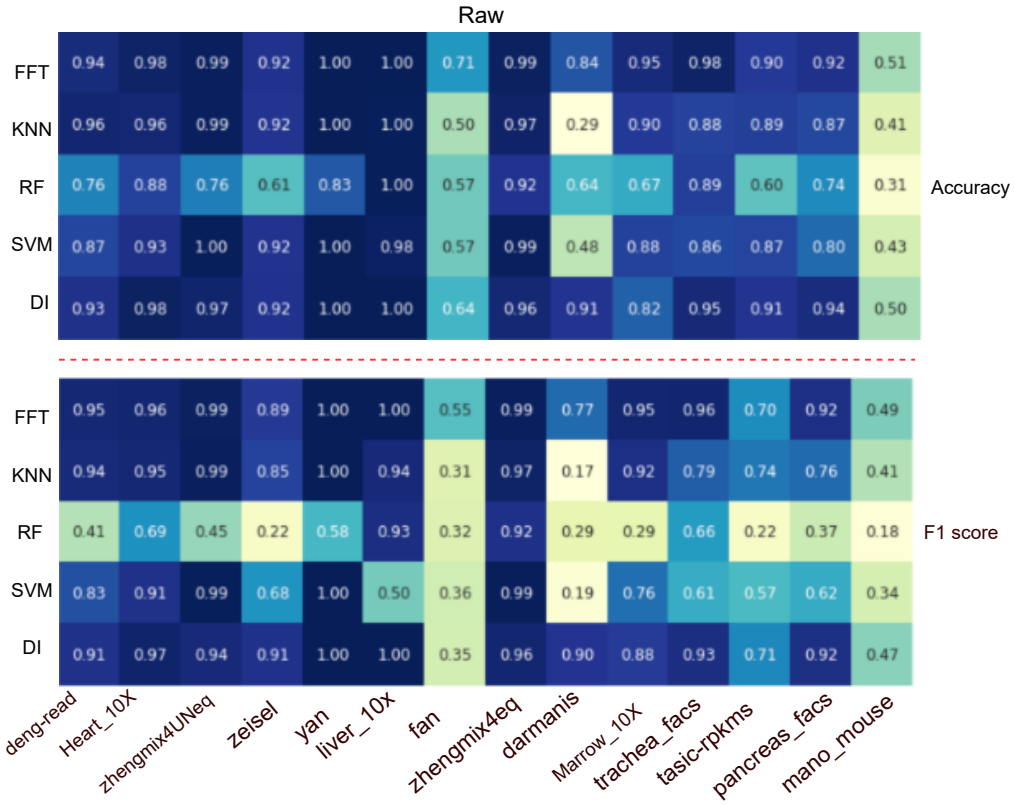

Fig. 3. Test accuracy and F1 score of machine learning models (KNN, RF, SVM) and of the CNN model on two different imagification methods (FFT and DeepInsight) across 14 datasets with omics raw data

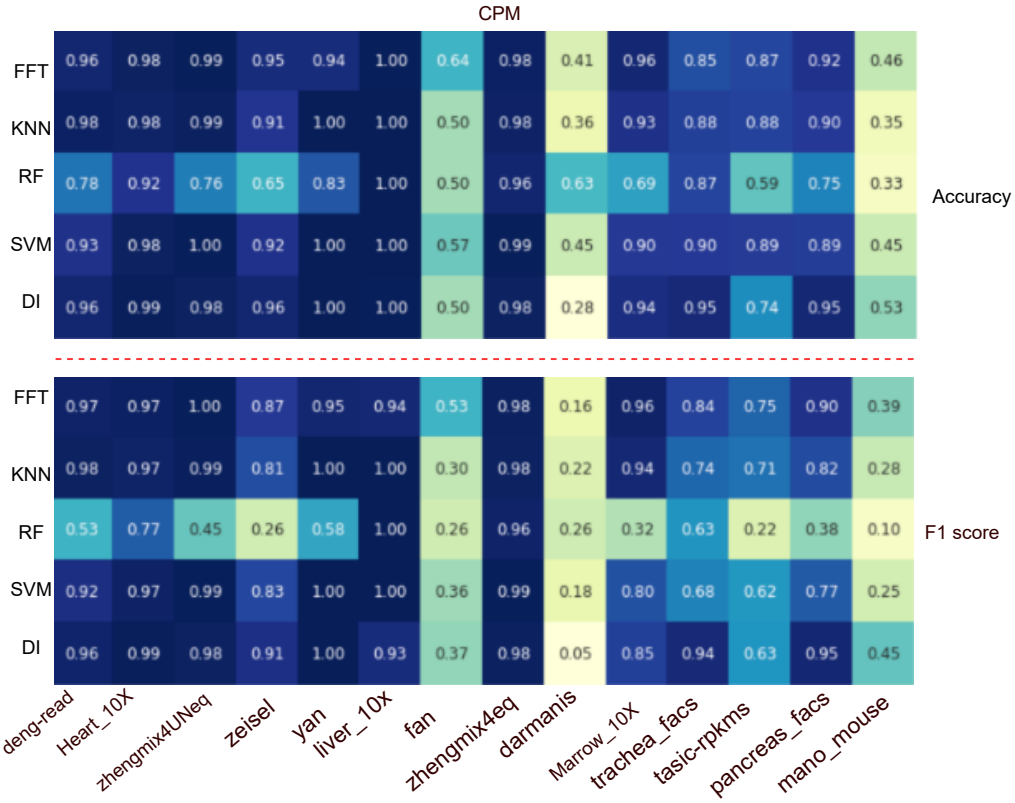

Fig. 4. Test accuracy and F1 score of machine learning models (KNN, RF, SVM) and of the CNN model on two different imagification methods (FFT and DeepInsight) across 14 datasets with omics data pre-processed through counts per million, CPM

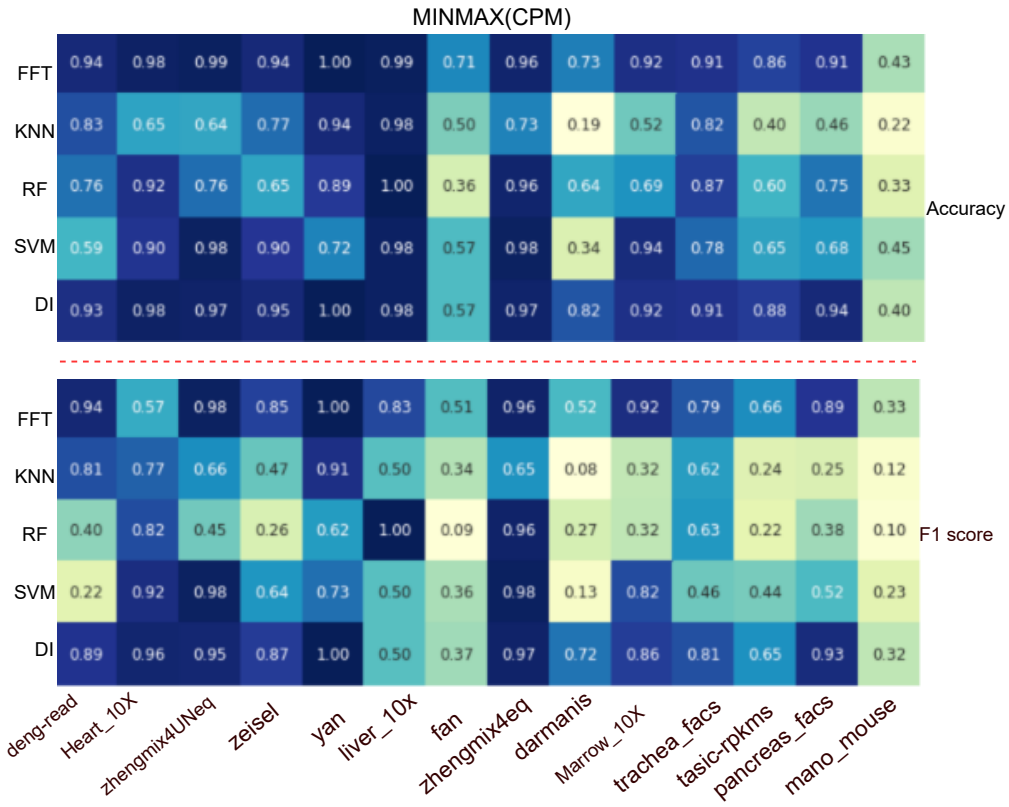

Fig. 5. Test accuracy and F1 score of machine learning models (KNN, RF, SVM) and of the CNN model on two different imagification methods (FFT and DeepInsight) across 14 datasets with omics data pre-processed through MinMax scaling over CPM

In figure 6, 7 and 8, three horizontal box-plots of Accuracy and F1 score with independent normalization across five models are displayed. The box plot is a valuable statistical technique for uncovering hidden patterns in a set of numbers, which provides a visual summary and comparison for multiple data points, and also shows its ability to easily identify outlier data. The transformed image based FFT model significantly outperform other traditional classifiers, whereas the DeepInsight method and the FFT model are evenly matched. In comparison, the FFT boxplot is very short, with a good F1 score and accuracy across multiple datasets and normalization methods. Nevertheless, KNN, SVM and especially RF perform inadequately in the classification task of single-cell spanning several datasets.

To further evaluate the performance of the proposed model,a confusion matrix of transformed images on ”Deng-read”,”yan”, ”pancrease-FACS” and ”zeisel” for CNN across test data is shown in Figure 9. As clearly seen, the CNN model utilizing the FFT image transformation method performs greatly on classification of cell types with small populations, so the results demonstrate the capacity of FFT image-form to enhance rare cell type recognition.

In table II, we calculated the p-value to show the differences among different models on accuracy and F1 score. P values measure how compatible our experimental data are with the null hypothesis [35]; it assesses how strongly the data support the premise that the null hypothesis is correct. Thus, FFT image-form method on CNN-model can significantly achieve

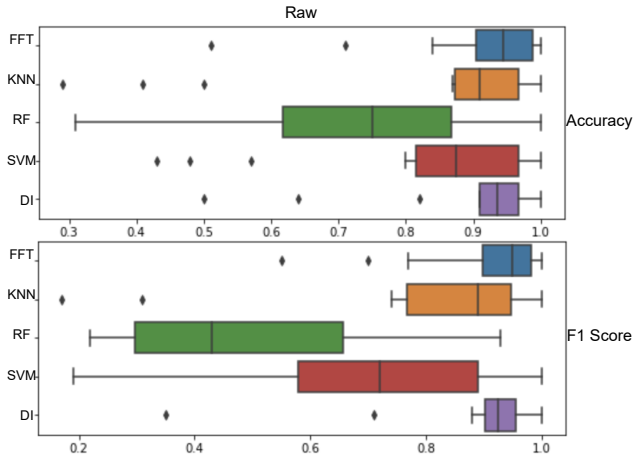

Fig. 6. Box-plot of Accuracy and F1 score on RAW data across 5 models

better performance in comparison to the best-performing scRNA-seq classifiers such as SVM, KNN and RF. The FFT method’s results are comparable with DI model, however, FFT method can significantly compete DI in computational time. To further compare the performances as well as CPU times of these models, we used an immensely large dataset which is a collection of several datasets. The combined dataset contains **11986** samples, **23433** features and **120** classes, which is way larger than the previous 14 datasets we have experimented. With a massive dataset like this, both models will be pushed

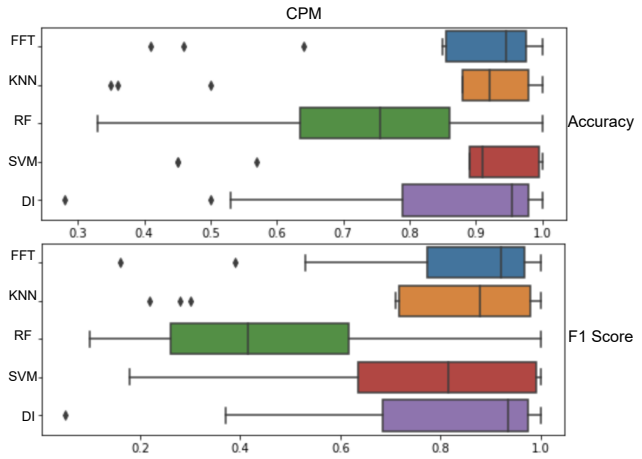

Fig. 7. Box-plot of Accuracy and F1 score on CPM data across 5 models

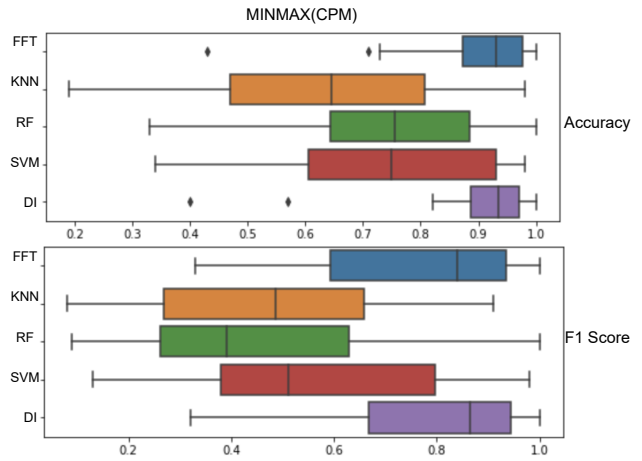

Fig. 8. Box-plot of Accuracy and F1 score on MINMAX(CPM) data across 5 models

to their limits, which offers us an unique perspective on how effective the FFT model is. See table III.

It is impressive to see that both models continue to yield high performances on a large dataset, however, FFT method outperforms DI. Moreover, it takes DI 18 minutes and 29 seconds to transform one dimensional raw data into two dimensional images in CPU time, whereas FFT model can complete the entire transformation process in 11 minutes and 40 seconds. The FFT model is 37% more efficient than DI in the transformation process. The backbone of DI involves dimensionality reduction method like t-SNE. DI in t-SNE step carries out the similarity of various elements and group similar elements together and push distinct elements further apart.

TABLE II
P-VALUES OF DIFFERENT MODELS ON ACCURACY AND F1 SCORE

	FFT					
	RAW		CPM		MNX(CPM)	
	Acc	F1	Acc	F1	Acc	F1
KNN	0.3007	0.2175	0.8179	0.7484	0.0032	0.0041
RF	0.0076	0.00003	0.1195	0.004	0.0404	0.0052
SVM	0.2472	0.0157	0.9776	0.572	0.0759	0.4489
DI	0.791	0.7763	0.8971	0.8821	0.946	0.9661

TABLE III
ACCURACY AND CPU TIMES ON THE COMBINED DATASET

Model	Accuracy	CPU Time
FFT	84.29%	11min 40s
DeepInsight (DI)	79.25%	18min 29s

Such computation is very time consuming compared to the FFT model using Fourier transformation calculation and easy to scale.

The novelty of the FFT model is reflected in the core transformation mechanism. A majority of existing image transformation techniques involve dimensionality reduction such as t-SNE [23] and PCA [12]. The FFT image-formed model can extract necessary information from 1D vector and perform image transformation without using any dimensionality reduction methods. The essence of this method is to carry out real and imaginary parts of Discrete Fourier Transform (DFT) [36] as sinusoidal space for 2D image conversion. This decomposes a data into constituent spectral components and hence yield frequency information about the sequencing data.

In summary, the FFT conversion technique establishes a connection between non-image data and image data. This promotes comprehensive analysis of numerous 1D medical data while employing the potential of CNN classifiers such as Tumor type classification [11], feature extraction of gene-expression [14] and breast cancer diagnosis [17].

VI. Conclusion

Depending on the underlying technology used, various omics profiles can include measurements of several hundreds to several thousands of molecules. The term *omics imagification* is proposed in this paper as a process of converting a vector representing these numerical measurements into an image with a one-to-one relationship to the corresponding sample. The proposed imagification approach transfers gene-expression vectors to two-dimensional images to classify the sample phenotype using automated image-recognition technology. The image-formed data conveys more information about cell-types into the classifier, including gene expressions and gene intensities. The proposed method was applied to single-cell RNA sequencing (scRNA-seq) datasets for evaluation. The experimental results demonstrate that a simple convolutional neural network (CNN) architecture can significantly outperform commonly-used machine learning techniques such RF, KNN, and SVM.

VII. Data and Code Availability

All the datasets used in this study are public and accessible from Gene Expression Omnibus (GEO). The entire code base including the python implementation of the proposed method and compared techniques are available at `https://github.com/VafaeeLab/omics-imagification`.

References

[1] A. Conesa, S. Beck, Making multi-omics data accessible to researchers, Scientific data 6 (1) (2019) 1–4.

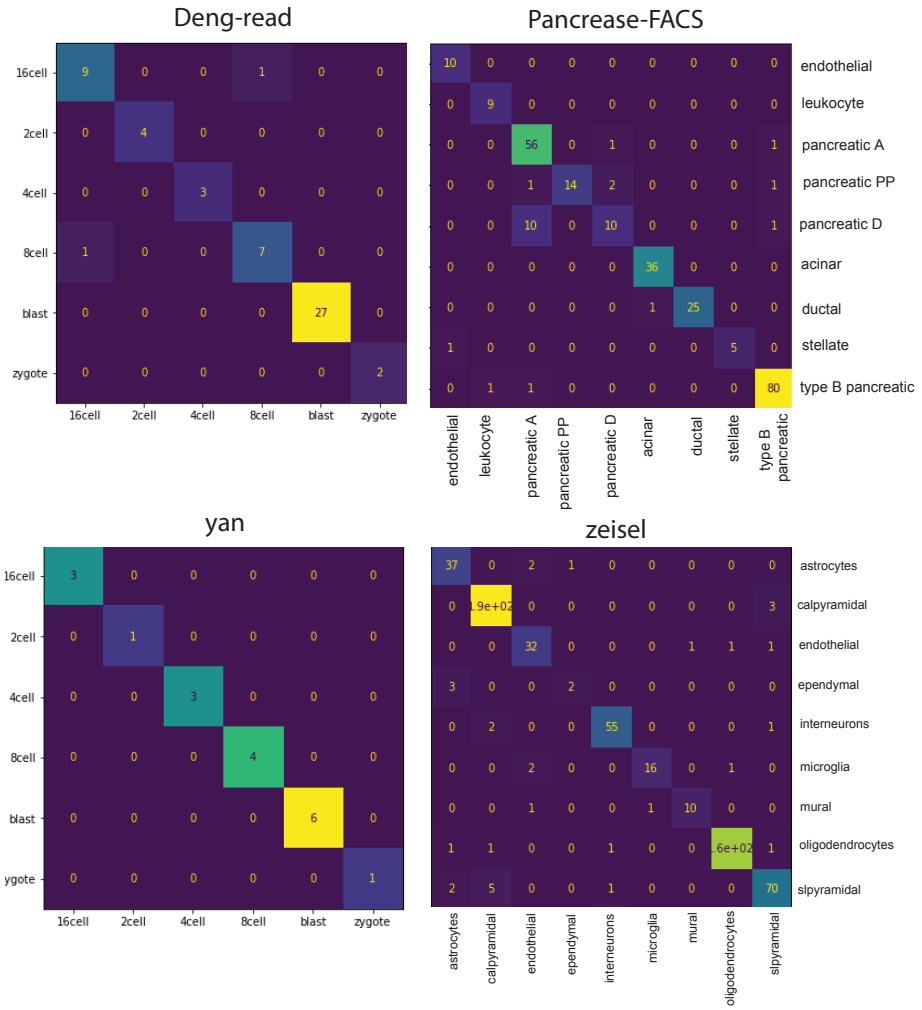

Fig. 9. Confusion matrix of transformed images on test data for CNN

[2] A. A. Kolodziejczyk, J. K. Kim, V. Svensson, J. C. Marioni, S. A. Teichmann, The technology and biology of single-cell rna sequencing, Molecular cell 58 (4) (2015) 610–620.

[3] M. Krzak, Y. Raykov, A. Boukouvalas, L. Cutillo, C. Angelini, Benchmark and parameter sensitivity analysis of single-cell rna sequencing clustering methods, Frontiers in genetics (2019) 1253.

[4] T. Abdelaal, L. Michielsen, D. Cats, D. Hoogduin, H. Mei, M. J. Reinders, A. Mahfouz, A comparison of automatic cell identification methods for single-cell rna sequencing data, Genome biology 20 (1) (2019) 1–19.

[5] O. Franzén, L.-M. Gan, J. L. Björkegren, Panglaodb: a web server for exploration of mouse and human single-cell rna sequencing data, Database 2019 (2019).

[6] S. Lall, A. Ghosh, S. Ray, S. Bandyopadhyay, sc-renf: An entropy guided robust feature selection for single-cell rna-seq data, Briefings in Bioinformatics 23 (2) (2022) bbab517.

[7] S. M. Zandavi, F. C. Koch, A. Vijayan, F. Zanini, F. Mora, D. Ortega, F. Vafaee, Disentangling single-cell omics representation with a power spectral density-based feature extraction, Nucleic Acids Research (05 2022). doi:10.1093/nar/gkac436.
URL https://doi.org/10.1093/nar/gkac436

[8] Y. Bengio, Learning deep architectures for AI, Now Publishers Inc, 2009.

[9] E. O. Brigham, The fast Fourier transform and its applications, Prentice-Hall, Inc., 1988.

[10] S. Weinstein, P. Ebert, Data transmission by frequency-division multiplexing using the discrete fourier transform, IEEE transactions on Communication Technology 19 (5) (1971) 628–634.

[11] B. Lyu, A. Haque, Deep learning based tumor type classification using gene expression data, in: Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics, 2018, pp. 89–96.

[12] J. Yang, J.-y. Yang, From image vector to matrix: A straightforward image projection technique—impca vs. pca, Pattern Recognition 35 (9) (2002) 1997–1999.

[13] B. Kovalerchuk, B. Agarwal, D. C. Kall, Solving non-image learning problems by mapping to images, in: 2020 24th International Conference Information Visualisation (IV), IEEE, 2020, pp. 264–269.

[14] G. López-García, J. M. Jerez, L. Franco, F. J. Veredas, Transfer learning with convolutional neural networks for cancer survival prediction using gene-expression data, PloS one 15 (3) (2020) e0230536.

[15] S. Anders, Visualization of genomic data with the hilbert curve, Bioinformatics 25 (10) (2009) 1231–1235.

[16] B. Moon, H. V. Jagadish, C. Faloutsos, J. H. Saltz, Analysis of the clustering properties of the hilbert space-filling curve, IEEE Transactions on knowledge and data engineering 13 (1) (2001) 124–141.

[17] A. Sharma, D. Kumar, Classification with 2-d convolutional neural networks for breast cancer diagnosis, arXiv preprint arXiv:2007.03218 (2020).

[18] S. M. Almufti, Historical survey on metaheuristics algorithms, International Journal of Scientific World 7 (1) (2019) 1.

[19] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Ieee, 1995, pp. 39–43.

[20] A. Sharma, A new optimizing algorithm using reincarnation concept, in: 2010 11th International Symposium on Computational Intelligence and Informatics (CINTI), IEEE, 2010, pp. 281–288.

[21] D. E. Goldberg, J. H. Holland, Genetic algorithms and machine learning (1988).

[22] A. Sharma, E. Vans, D. Shigemizu, K. A. Boroevich, T. Tsunoda, Deepinsight: A methodology to transform a non-image data to an image for convolution neural network architecture, Scientific reports 9 (1) (2019) 1–7.

[23] L. Van der Maaten, G. Hinton, Visualizing data using t-sne., Journal of machine learning research 9 (11) (2008).

[24] A. Sharma, A. Lysenko, K. A. Boroevich, E. Vans, T. Tsunoda, Deepinsight-fs: Selecting features for non-image data using convolutional neural network, bioRxiv (2020).

[25] P. Heckbert, Fourier transforms and the fast fourier transform (fft) algorithm, Computer Graphics 2 (1995) 15–463.

[26] D. Singh, B. Singh, Investigating the impact of data normalization on classification performance, Applied Soft Computing 97 (2020) 105524.

[27] V. Y. Kiselev, A. Yiu, M. Hemberg, scmap: projection of single-cell rna-seq data across data sets, Nature methods 15 (5) (2018) 359–362.

[28] C. A. Vallejos, D. Risso, A. Scialdone, S. Dudoit, J. C. Marioni, Normalizing single-cell rna sequencing data: challenges and opportunities, Nature methods 14 (6) (2017) 565–571.

[29] P. V. Kharchenko, L. Silberstein, D. T. Scadden, Bayesian approach to single-cell differential expression analysis, Nature methods 11 (7) (2014) 740–742.

[30] N. L. Bray, H. Pimentel, P. Melsted, L. Pachter, Near-optimal probabilistic rna-seq quantification, Nature biotechnology 34 (5) (2016) 525–527.

[31] P. Juszczak, D. Tax, R. P. Duin, Feature scaling in support vector data description, in: Proc. asci, Citeseer, 2002, pp. 95–102.

[32] J. M. Keller, M. R. Gray, J. A. Givens, A fuzzy k-nearest neighbor algorithm, IEEE transactions on systems, man, and cybernetics (4) (1985) 580–585.

[33] M. Pal, Random forest classifier for remote sensing classification, International journal of remote sensing 26 (1) (2005) 217–222.

[34] W. S. Noble, What is a support vector machine?, Nature biotechnology 24 (12) (2006) 1565–1567.

[35] D. R. Anderson, K. P. Burnham, W. L. Thompson, Null hypothesis testing: problems, prevalence, and an alternative, The journal of wildlife management (2000) 912–923.

[36] S. Winograd, On computing the discrete fourier transform, Mathematics of computation 32 (141) (1978) 175–199.