

An End-to-End Breast Tumour Classification Model Using Context-Based Patch Modelling- A BiLSTM Approach for Image Classification

Suvidha Tripathi^{1,*}, Satish Kumar Singh¹, and Hwee Kuan Lee²

¹Department of Information Technology, Indian Institute of Information Technology Allahabad, Devghat, Jhalwa, Prayagraj-211015, India

²School of Computing, National University of Singapore, 13 Computing Drive, 117417, Singapore, Bioinformatics Institute, A*STAR, 30 Biopolis Street, 138671, Singapore,

Image and Pervasive Access Lab(IPAL), CNRS UMI 2955, 1 Fusionopolis Way, 138632, Singapore, Singapore Eye Research Institute, 20 College Road, 169856, Singapore

*Corresponding author: Suvidha Tripathi, suvitri24@gmail.com

Abstract

Researchers working on computational analysis of Whole Slide Images (WSIs) in histopathology have primarily resorted to patch-based modelling due to large resolution of each WSI. The large resolution makes WSIs infeasible to be fed directly into the machine learning models due to computational constraints. However, due to patch-based analysis, most of the current methods fail to exploit the underlying spatial relationship among the patches. In our work, we have tried to integrate this relationship along with feature-based correlation among the extracted patches from the particular tumorous region. For the given task of classification, we have used BiLSTMs to model both forward and backward contextual relationship. RNN based models eliminate the limitation of sequence size by allowing the modelling of variable size images within a deep learning model. We have also incorporated the effect of spatial continuity by exploring different scanning techniques used to sample patches. To establish the efficiency of our approach, we trained and tested our model on two datasets,

microscopy images and WSI tumour regions. After comparing with contemporary literature we achieved the better performance with accuracy of 90% for microscopy image dataset. For WSI tumour region dataset, we compared the classification results with deep learning networks such as ResNet, DenseNet, and InceptionV3 using maximum voting technique. We achieved the highest performance accuracy of 84%. We found out that BiLSTMs with CNN features have performed much better in modelling patches into an end-to-end Image classification network. Additionally, the variable dimensions of WSI tumour regions were used for classification without the need for resizing. This suggests that our method is independent of tumour image size and can process large dimensional images without losing the resolution details.

1 Introduction

Image based computational pathology has developed into an ever-evolving field for computer vision researchers. New methods are being introduced frequently in this field for natural everyday scenes, face recognition, video analysis and other forms of biometrics. Despite that, the rate of development of medical image CAD algorithms for enhancing their diagnostic performance could not mirror the rate of development of new natural scenes analysis algorithms. It may have been due to the highly heterogeneous nature of cancer cells, which increases the complexity of the task at hand. In context with breast cancer, extensive research based on oncogenic pathways and tumor cell metabolism, and based on chemotherapeutic observations, it has been realized by pathologists that the disease is quite unpredictable [1].

Hence, there is a pressing need for the development of computer vision algorithms that are particularly advanced for diagnostic and prognostic evaluation of digitized biopsy images. Until then, there is a progressive adaptation of currently available state of the art methods for cancer detection, segmentation, and classification. The importance of precise prognosis in this field requires the differentiation of digitized samples into two, three, or more classes. In our work, we have four classes, Normal, Benign, *In situ* carcinoma, and Invasive Carcinoma. The process of classification of breast samples by the pathologist help in a more accurate understanding of the disease and consequently help in the directed treatment of patients. The manual process is; however, quite a time consuming and requires an expert’s knowledge due to the underlying complexity of the nature of images. The efforts to automate such non-trivial problem requires expert intervention to verify the diagnosis made by the CAD process. Besides that, the fea-

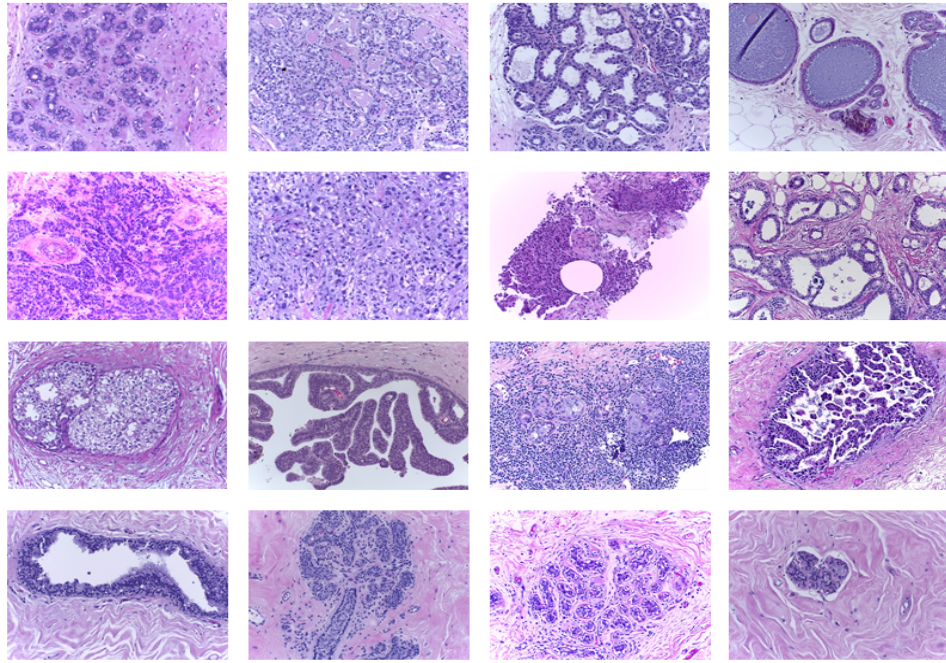


Figure 1: Microscopy BACH data samples. First row to fourth: Benign Tumours, Invasive Carcinoma, *In situ* Carcinoma, and Normal

sibility of implementation of such algorithms poses preliminary challenges. For instance, high resolution of gigapixel Whole Slide Images could not be processed by any current state of the art algorithms due to their large size. A large amount of information present in one patient slide makes the task more challenging concerning space and efficiency. Therefore, for practical problem solving, we need to either build new systems that could address such challenges or find a workaround of our problems that could be feasibly addressed by available systems. One such workaround is dividing WSI into patches of the size that could be easily fed into the algorithm. However, this leads to loss of overall structure of the tumor and various other sub-structures present in the slide. The spatial continuity of the patches also becomes hard to incorporate within a deep end-to-end model. The task becomes more non-trivial in case of a 4-class problem rather than a 2-class problem where the structures need to be segregated between two widely spaced classes. As the number of classes or segregation increases, the space between classes reduces.

Considering all these issues, we chose our model such that the gigapixel size of WSI could be harnessed in a way without losing the structure of the overall suspected region. The spatial relationship between the patches of the same region could be modeled end-to-end without the need of building a separate algorithm to infuse the context of the previous patch in the sequence of the patches which together makes an entire tumor region.

One such state of the art model which is well known for preserving the contextual relationship, is Recurrent Neural network, commonly known as RNN. RNNs have been used by the computer vision community to process sequences such as texts and videos. We acknowledged its efficiency and formulated our problem around the strength of the RNNs, which is processing sequences of patches from the same region and eventually classify the input sequence as one of the four classes. We classified image regions as a whole using BiLSTMs. BiLSTMs are a known version of RNNs for modeling textual and video sequences. They have been widely used for activity recognition in videos and have proved their niche in modeling future contextual information due to their bi-directional architecture. Our method could serve its purpose in clinical diagnosis by assisting pathologists for labeling suspected regions automatically. Our main contribution is summarised in following points:

1. According to our knowledge, this is the first study that includes the use of contextual information among the patches from the same region using BiLSTMs for classification of tumors.

2. Our method is robust to the size of the tumour regions as it can take both very huge dimensions WSI and microscopy regions. In this study, the range of tumour regions vary between 17290 to 236 pixels across height and 20570 to 195 pixels across the width.
3. The study did not alter the size of the tumors for deep modeling and classify variable size tumor regions by processing them as a sequence of features.
4. This work proposed end-to-end network for patch to image classification unlike previous literatures that use stage wise networks to first classify patches and then aggregate classification results of patches into image labels [2, 3, 4, 5, 6, 7, 8, 9].
5. This is a shallow network that do not require heavy training to train hundreds of layers as in ResNet and GoogleNet.
6. We also experimented with patch scanning methods to verify that a particular scanning technique that deploy maximal connectivity between patches is better than randomly extracting patches from the image.

2 Related Work

The application of RNN based architectures such as LSTM [10] and BiLSTMs [11, 12] on series data classification such as texts and time series has been a very common methodology. Researchers have recently started combining CNNs and LSTMs for image captioning [13, 14, 15] or multi-label image classification [16, 17, 18, 19] as well. The idea of using RNN based models for image classification stemmed from the fact that objects in an image are often, though not always, related to each other in some way. Images, although, are not sequential data but carry some latent semantic dependencies which can be modeled as a sequence of occurrences of certain objects present in the image that overall define the global image description. These deep LSTMs based models have, however, are not sufficiently explored on high-resolution medical data. With high dimensional images in case of WSIs, the tumor regions when divided into patches can act as a sequential data that have some contextual dependency with each other. Modeling this contextual information among patches is a crucial step to perform slide level classification.

There have been studies in Whole Slide Image level analysis that have drawn contextual and spatial relationship among patches using their novel methods. For instance, authors in [7] proposed a deep spatial fusion network to predict image-wise label from patch-wise probability maps. They evaluated their network performance on two datasets BIC [20] and BACH [21] and used heavy augmentation due to the small volume of images. Their network was not end-to-end and required heavy data pre-processing steps to enhance the performance. They used microscopy images to test their model, which have dense class properties. Whereas, in the case of Whole Slide Image annotations, the tumour class like Invasive carcinoma could be spread across the gigapixel image and the parts of the annotation may look like normal. Therefore, with WSIs, the parts of the annotation when broken into patches, may not give the reliable label. Hence, such methods should be tested on such datasets as well for better clinical significance. The method in [8] exploits the spatial context between patches extracted from high resolution histopathological images for grading of colorectal cancer histology images. The authors propose a two staged framework consisting of two stacked CNNs. The first CNN called as LR-CNN learns the representations of the patches and aggregates the learned features from each patch in the same spatial dimension as the original image ($M \times N$). So, in other words LR-CNN converts a high resolution image into high dimensional feature map. The next stage consists of context aware blocks called as RA-CNN that takes feature representation cube as input to learn the spatial relationship between patches to make a context-aware prediction. The authors explored different network architectures for context-aware learning. The strategy solves a huge challenge of missing contextual information in patch-based classifiers. The robustness of the method also lies in the fact that the use of pre-trained architectures to extract features reduces the time and effort to train large models. However, the authors did not test their method on WSIs which pose a challenge of multi-resolution feature learning and very large size. In case of WSIs the feature cube could be as large as high resolution images and then its processing in a deep learning network could become infeasible.

To address the problem of multi-resolution analysis, majority of the previous works of literature have used patch-level analysis which requires breaking up of structures and hence global level features are lost. But, due to multi-resolution data, it is in fact left as an only choice to process such images. All the methods using WSI datasets discussed above have done the same for developing their models. Studies like [2, 3, 4, 5, 6] have performed patch-based modelling of histopathology slides or microscopy images

to perform image-wise classification using methods like probability fusion and majority voting. The authors in [5] developed a two-stage processing pipeline for classifying WSIs of gastric cancer. The first stage- discriminative instance selection selected the most informative patches on the basis of probability maps generated by a localization network. The second stage performed the image level prediction. The authors proposed a novel recalibrated multi-instance deep learning network (RMDL) with the purpose of aggregating both local and global features of each instance via a modified local-global feature fusion module. RMDL framework presented an effective way to aggregate patches for final image level prediction by exploiting the interrelationship of the patch features and overcame the drawbacks of direct patch aggregation. The method is however limited in its approach as it is confined to same scale context and do not address the spatial relationship between the instances.

The authors in [22] studied the applicability of deep learning architectures in identifying the breast cancer malignant tumours from benign tumours. The different sets of experiments were designed to train the CNN with different strategies that allow both high and low resolution images as input.

In [23] two CNN architectures have been used to identify breast cancer tumour and the magnification of the image. Single Task CNN classifies the benign and malignant tumour. Whereas, multi-task CNN has two output branches which takes multi-resolution image patches as input and produces two classification – between malignant and benign and between four classes of magnification.

Similarly, Araujo et al. [9] first proposed a patch-wise classification and then combined the patch probabilities to perform image-wise classification. They used their custom CNN model to perform patch-wise classification and achieved 66.7% accuracy. Then the majority voting scheme was used among the classified patches to predict the overall image label. This method was also not end-to-end and required extensive CNN training and experiments to decide optimal hyper-parameters for their proposed model. They also did not consider spatial context among the patches to build a relationship between same image patches which may have proved crucial performance enhancer.

All these methods although solve the challenge of multi-resolution analysis by patch-level aggregation of classification results, suffer from lack of spatial context and continuity relationship among patches. Moreover, due to the inherent limitation of state-of-the-art deep learning models which takes only a fixed size input, the previous works of literature had to sometimes

perform heavy resizing to conform to the size of network input. Therefore, CNN + RNN based model could be the perfect replacement of such models since they could provide both spatial and contextual modelling, strategic region extraction method without the limitation of resizing along with the end-to-end compact model to process high-resolution Whole Slide and microscopy images.

Few of the recent pieces of literatures have used such type of CNN + RNN models for the analysis of histopathological data. For instance, the paper [24] explores the application of deep reinforcement learning in predicting the diagnostically relevant regions and their HER2 scores in breast immunohistochemical (IHC) Whole Slide Images. For the given task, the authors proposed context module and a CNN-LSTM end-to-end model. The model intelligently views the WSI as the environment and the CNN-LSTM acts as a decision maker or the agent. Their model successfully mimics the histopathological expert analysis that first looks coarsely at ROIs at low resolution and then predict the scores of diagnostically relevant regions. Their model also incorporates multi-resolution analysis by combining features of the same region at multiple resolution for better predictive performance. The main advantage given by their model is that one need not look at all the regions of a WSI to predict the outcome and instead could focus on small number of regions without sacrificing the performance of the model. Similarly, [25] and [26] have also used the combination of CNN-LSTM for disease outcome prediction. The authors [25] have used the genomic data (Pathway Scores PS) with disease recurrence extracted from gene expression signatures exhibited in prostate tumors with a Gleason 7 score to identify prognostic marker. They calculated the PS scores and combined them with deep learning model for the purpose of combining the prognostic markers with image biomarkers. The deep learning model used is CNN-LSTM end-to-end model that take WSI patches as input sequence. CNN finds the features which LSTM processes to output the final hazard ratios of recurrence of the disease. They compared their model performance with different image features (LBP, HOG, SURF, neurons) with pathway scores. The results shows higher hazard ratios with CNN-LSTM + PS in comparison to other clinically relevant prognostic features used in the comparison. The model show a novel idea of combining genetic markers with image biomarkers using LSTM in their model in order to preserve the spatial and contextual relationship among patches. However, the model is not sufficiently validated with different datasets along with their choice of CNN model and choice of training parameters. The paper [26] predicts the five-year disease specific survival of patients diagnosed with colorectal cancer directly from digitized

images of haematoxylin and eosin (H&E) stained diagnostic tissue samples. The authors used a CNN-LSTM based model that takes TMA spots as input sequence into the model. The VGG16 architecture was used to extract patch features. The model claims the novelty of providing direct outcome prediction instead of doing intermediate analysis like classifying tissue samples. The proposed model by the authors used different scanning techniques to extract patches but claimed to have found no effect on the final prediction results. This claim is not properly validated in the study and is contradictory to what we found in our experimental analysis. The authors compared their model with traditional machine learning classifiers such as naïve-bayes, logistic regression, SVM. The lack of comparison with contemporary deep learning classifiers weakens the validation of the proposed method. All these methods using CNN-LSTM as their base model have shown the applicability of RNN based models in disease prognosis. Keeping the advantages in mind, we used the BiLSTMs, the Bidirectional LSTM to classify tumour regions in our work. The experimental observations on our dataset (Section 4.4) showed the advantage of using BiLSTMs over LSTMs in our model.

3 Methodology

3.1 Overview

In medical images, patch level classification is often useful for detecting cancer in microscopy and WSI images. However, if the prediction needs to be made for a whole tumor or gland, the network model needs to be trained such that the whole tumor region could be classified without losing its structure, resolution, and spatial correlation. For building such model, we have first extracted annotated tumour regions from WSIs and performed rotation transformation on regions for rotation invariance. After pre-processing of WSI dataset, we divided both microscopy and WSI tumour samples into patches. The patches were acquired by following different scanning techniques. We further developed a BiLSTM network model that takes the patches acquired from large tumour regions in the form of sequences. Since, the patches were extracted in a continuous pattern, therefore, we were able to construct a sequential data fit for BiLSTM network. We extracted features from each patch in a sequence using GoogleNet (pre-trained on ImageNet). Accumulated features per region formed one sequence. The sequences were then passed through BiLSTM layers for classification into labels. At the test time, the test regions follow the same feature extraction and sequence formation procedure. The trained BiLSTM model then tests the sequence

and give out the predicted label. In brief, the method follows the 5 steps: 1) extracting whole regions (Benign, Invasive, and *In situ*), 2) extracting patches from each tumor region, 3) extracting features from each set, per patch, 4) forming a sequence out of each set, and 5) sequence processing and classification.

3.2 Preprocessing

3.2.1 Region Extraction

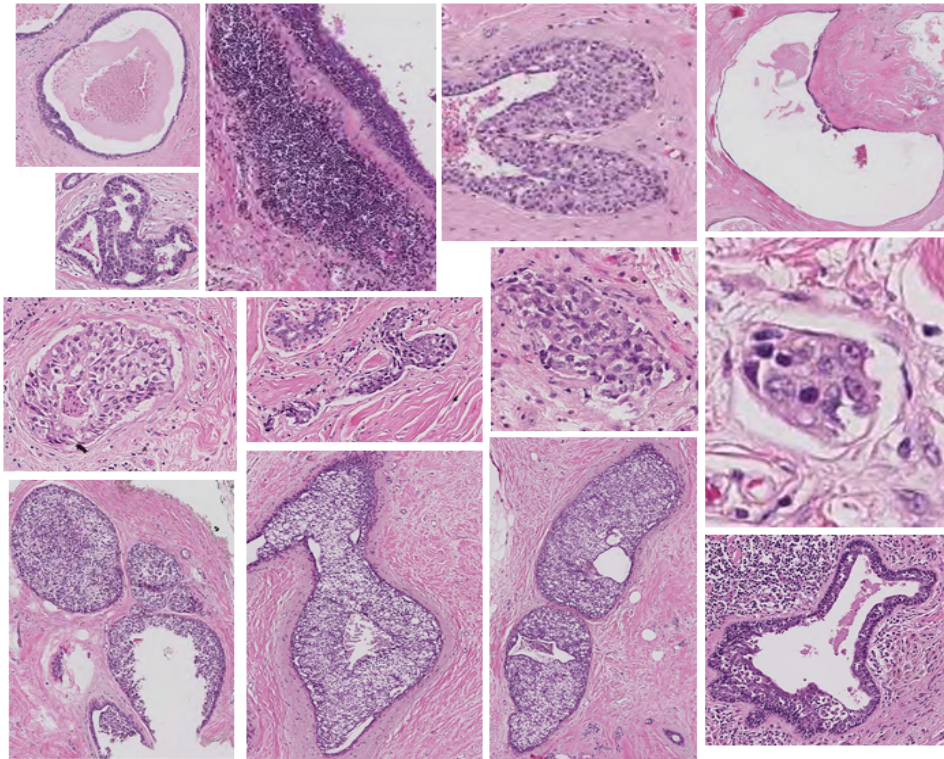


Figure 2: WSI BACH data samples extracted from gigapixel slides. The variable size of each tumour region pose limitation in traditional deep learning framework. But, our model mitigate this limitation by allowing variable sequence size. First row to third: Benign Tumours, Invasive Carcinoma, *In situ* Carcinoma. These regions can be seen having different dimensions but represent a single resolution level (level 0) from the WSI pyramid (Fig. 5)

The histopathological breast cancer slide dataset used in our work con-

tains ten annotated WSIs labeled into four major classes, Normal, Benign, *In situ* carcinoma, Invasive carcinoma. The annotation of each WSI is recorded in XML files. Each XML file is divided into regions as annotated by pathologists in the corresponding WSI. The regions are then marked by drawing a rough boundary around the suspected region. The boundary is marked using slide annotation tools such as ASAP (Automated Slide Annotation Platform). Each pixel coordinate annotated by the pathologist is recorded in the XML file under a current region being annotated. The XML file also contains the region label, area of the region in pixels, region id, zoom ration, length of the region in microns, and area of the region in microns. Each annotated coordinate is represented in X, Y, and Z axes values. From the available information, we calculated the maximum and minimum boundary coordinates to find out the location, height, and width of the labeled region.

Since the tumour regions can be found in varying orientations depending upon the angle of acquisition of the particular WSI or microscopy image, the model should be robust to such changes. Therefore, to make the process more robust and rotation invariant, the obtained regions were rotated by following a unified method. To determined the angle of rotation for a particular region, the region mask was used to analyse the orientation of the region with respect to the vertical axis. The angle of rotation was then calculated following the steps below:

1. Determine the major axis centroid of the region.
2. Calculate the major axis angle (M) from the X-axis.
3. Calculate the angle of rotation $R = 90 - M$
4. Rotate the region along the major axis centroid by the angle R .
5. Repeat steps 1 to 4 for both region and region mask
6. Calculate the bounding box coordinates of the rotated mask.
7. Modify the obtained bounding box dimensions to the nearest multiple of 256.
8. Crop rotated region around the modified bounding box coordinates.

3.2.2 Scanning Methods for Patch Extraction

Some of the extracted regions had large pixel dimensions due to their high resolution, which required breaking regions into patches to enable the processing of the regions. The arbitrary dimensions of the sampled regions was

also an issue for the deep network training since such networks require equal size images as input. Therefore, for the feasibility of the experiment, the regions were divided into patches of dimension 256×256 . The particular patch size was chosen keeping in mind following points:

- The smaller patch size in the power of 2 is 128×128 . This patch size contains less details than a 256×256 patch.
- The larger patch size 512×512 and more (in the powers of 2) although would contain more details and context, but will impose computational constraints like expensive computation resources and time. This scenario would not be feasible in hospital implementation and integration of the CAD methods.
- The pre-trained deep learning models like GoogLeNet, ResNet, DenseNet, InceptionV3 take fixed size input ranging from 200 to 300 pixels across their width and height. hence, taking smaller or larger patch sizes would demand heavy resizing resulting in loss of information and details. Therefore, 256×256 patch size seemed appropriate for the proposed method. Many recent literatures like *Wang et al.*, *Chennaswamy et al.* in [27] have resized their patches to 256×256 and then resized them to 224×224 in order to process them with deep learning architectures like ResNet and DenseNet.

To study and analyze the effect of different scanning techniques for sampling patches from regions, we tested three different scanning methods. Fig. 3 shows the pictorial representation of these techniques.

The first technique deploys most commonly used scanning method that moves the sliding window of desired patch dimensions from left to right across the width until the maximum width. The process is repeated across the height of the region. The window is non-overlapping, and at the extreme ends, if the expected height and/or width of the patch is greater than the remainder, we used symmetric padding to level the patch dimensions. For the convenience of the language, we addressed this scanning method as *Scan_1*. The process is illustrated in Fig. 3a.

The second scanning technique was thought as an attempt to arrange patches in sequence to bring as much continuity as possible. For any RNN method, where the sequence of data is the key to linking the context of the past and future with the present, we needed to derive sequential information from our tumor regions after they are sampled into patches. Our method is an effort to test the efficiency of RNN in case of image sequences. It scans

patches starting from left to the right across the width in one iteration, and then the second iteration starts from the next row of non-overlapping pixels. It starts from right towards left, covering the width of the image. The process is repeated for subsequent rows until the entire region is exhausted. We named this scanning technique as *Scan_2*, shown in Fig. 3b.

The third scanning method was deployed to bring more correspondence between the neighboring patches. The patches were scanned as represented in Fig. 3c. The set of four neighboring patches are scanned first, then the next adjacent batch, and onwards. When the row of non-overlapping pixels changes, the batches were scanned from right to left. The process was repeated until the region was covered across both dimensions. This technique is further referred to in the article as *Scan_3*. The patches from each region were separated in the form of sets or folders. Each folder contained an arbitrary number of patches according to the dimensions of the particular region. The patches in each folder were labeled the same as the label of the region.

The variable number of patches in each set does not limit the efficiency of our model. In fact, it allows huge dimensional tumour regions to be flexibly processed all at once in the form of a sequence without the need for heavy resizing. This flexibility pose as a strength of our model.

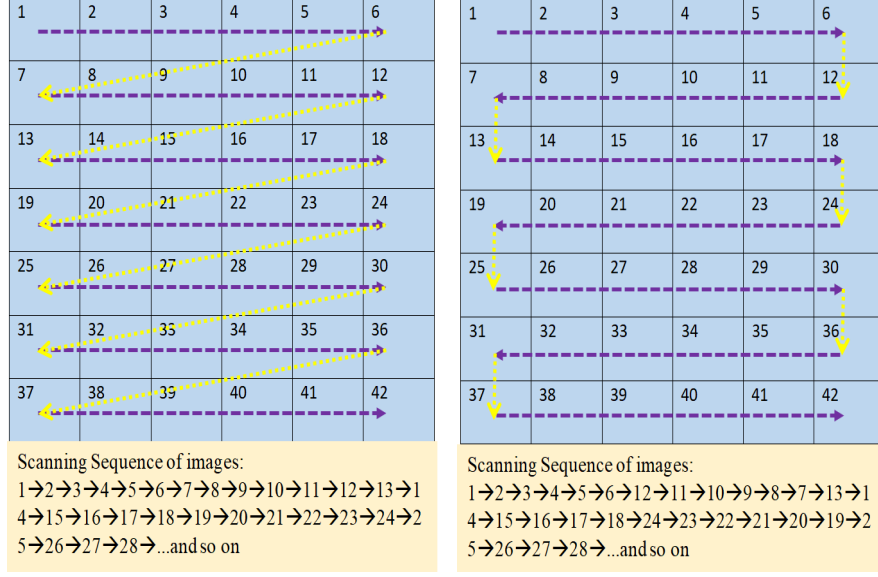
3.3 CNN Feature Extraction

After patch sampling process using all three scanning techniques, each set of patches was passed through pre-trained GoogleNet architecture available in MATLAB 2019a for feature extraction step. The GoogleNet architecture was not fine-tuned on our datasets, and hence the hefty training process was not required in our work. The simple pre-trained weights of this architecture were used to extract deep features from the patches, and the sequence of features was constructed from each folder to be processed by BiLSTM layers. The complete process is further elaborated in the subsequent section. For comparison purposes during the experimental analysis, we also used ResNet101 and DenseNet201 pre-trained architectures to show the performance effect on the final classification output.

3.4 Tumour Region Classification

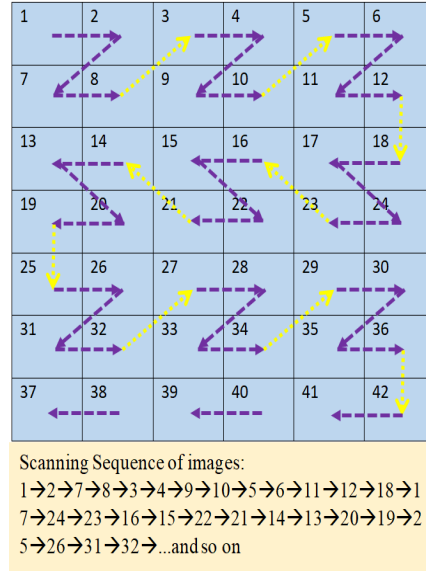
3.4.1 Patch Feature Sequence Formation

We used the GoogleNet pre-trained on ImageNet as a fixed feature extractor. The patches from each set are converted to sequences of feature vectors,



(a) Scan_1

(b) Scan_2



(c) Scan_3

Figure 3: The figure illustrates the different scanning methods that are used to extract patches from labeled WSI regions. The numbered blue blocks represents the patches in the WSI or Microscopy dataset. The dotted purple arrow shows the direction of scan and the dotted yellow arrow shows the transition from one pass of scan to another.

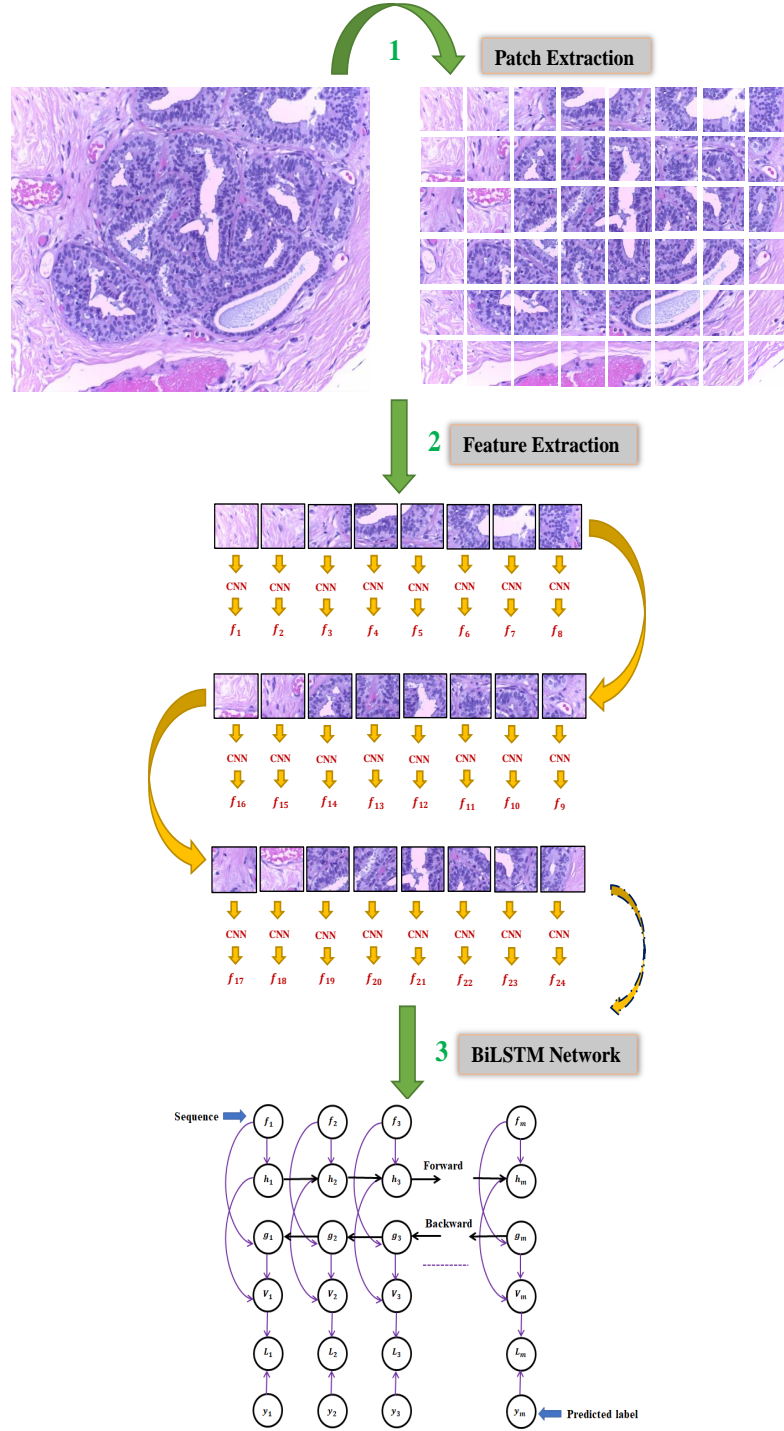


Figure 4: Illustration of the whole process pipeline from patch extraction to computation of a BiLSTM network. The steps are shown through green arrows. The first step is to extract patches, followed by CNN feature extraction and Sequence formation (f_1, f_2, f_3, \dots). The sequence of features is then used as a input to a BiLSTM network. BiLSTM network is meant to learn to map input sequences f to target sequences y . The h recurrence propagates information forward in time (towards the right), while the g recurrence propagates information backward in time (towards the left).

where the feature vectors are the output of the activations function on the last pooling layer of the GoogleNet network ("pool-7 \times 7_s1"). We have used the pre-trained network because we did not have much training data to train a network from scratch, and there were no standard pre-trained weights publically available on similar medical data. Each sequence is a D-by-m array, where D is the number of features (the output size of the pooling layer) and m is the number of patches in the region. Feature Dimension for one patch = 1024 X 1 (for GoogleNet features), feature dimension for m patches in a region = $1024 \times m$; Labels for m patches = label of the region. Let for patch 1 feature vector is f_1 , patch 2 : f_2 , patch 3 : f_3, \dots , Patch m : f_m , so the n th sequence comprise : $f_1, f_2, f_3, f_4, \dots, f_{m-1}, f_m$. Each labeled region in a WSI region forms a single sequence of patches. In other words, one region is converted into one sequence. We can then divide these sequences into training, testing, and validation sets.

3.4.2 BiLSTM Training and Classification

BiLSTMs or Bi-directional Long Short Term Memory models are different from traditional LSTMs by capturing information from past as well as future. This type of network is feasible for applications where prediction depends on the whole input sequence. BiLSTMs combine two LSTMs where one LSTM take input sequence from start to end while the other LSTM takes input sequence from end to the first patch in the sequence. Figure 4 illustrates the BiLSTM model, with $h_{(t)}$ is the state of the sub-BiLSTM that moves forward through the ordered sequence and $g_{(t)}$ represents the state of the sub-BiLSTM that moves backward through the sequence where $t = 1, 2, 3, \dots, m$. The output unit $V_{(t)}$ is obtained by concatenating $h_{(t)}$ and $g_{(t)}$. $V_{(t)}$ is a representation that depends on both the past and the future of the sequence but is most sensitive to the current inputs. An output vector $V_{(t)}$ is calculated as

$$V_{(t)} = f(h_{(t)}, g_{(t)}) \quad (1)$$

where function f is used to combine the two output sequences. It can be a concatenating function, a summation function, an average function or a multiplication function. The following vector can represent the final output of a BiLSTM layer,

$$V_m = f(h_m, g_m) \quad (2)$$

in which V_m , is the predicted sequence. Such a network where only the final output vector is sufficient to summarize a sequence is useful for predicting

the label of the patch sequence. To Train this network, the cross-entropy loss function L is used at the end to back-propagate information first through forward h states and second through backward states g . After forward and backward passes, the weights are updated. The sequence sets are passed through BiLSTM one at a time, and the predicted output tells the class of the sequence. We have used softmax classifier for our prediction. The end-to-end network architecture is described in the Table 1

Table 1: End-to-end architecture of the Tumour classification network.

Layer	Type	Input/Output Dimensions	Description
1	Sequence Input Layer	$224 \times 224 \times 3$	Enables sequence data input to a network
2	Sequence Folding Layer	Out: $224 \times 224 \times 3$ Minibatch: 1	This layer enables processing of a batch of sequence input as a batch of images
3-140	Convolution Layers (GoogleNet)	Input: $224 \times 224 \times 3$ Output: $7 \times 7 \times 1024$	All the middle layers of GoogleNet including convolution, ReLU, Batch Normalization, Dropout, etc.
141	Average Pooling Layer (pool5-7 \times 7.s1)	$1 \times 1 \times 1024$	Average the the input feature dimension (7×7) to (1×1)
142	Sequence Unfolding Layer	$1 \times 1 \times 1024$	Restores the sequential structure of the input sequence of images. Minibatch output of sequence folding layer is connected to minibatch input of this layer.
143	Flatten Layer	1024	Reshapes the 3 dimensional feature vector to one dimension
144	BiLSTM Layer (2000 hidden units)	4000	Enables learning bidirectional long term dependencies between sequence of patches from a region. Hidden units correspond to amount of information remembered between time steps or hidden states of BiLSTM
145	Dropout Layer	4000	Randomly sets input features to zero with a specified probability. This is added to prevent network overfitting.
146	Fully Connected Layer	3	Multiplies the weight matrix and adds bias to the input features.
147	Softmax Layer	3	Applies softmax function to the input
148	Classification Layer	—	Computes the cross-entropy loss.

4 Setup and Results

4.1 ICIAR 2018 BACH Dataset

The BACH (**B**re**A**st **C**ancer **H**istology) dataset was released by ICIAR 2018 conference organizers as a grand-challenge for classification and localization of tumors segregated by clinically relevant four classes. The dataset was released in two parts, microscopy and whole slide images. The microscopy image dataset contains 400 histology images, each with dimensions 2048×1536 . The 400 microscopy images are subdivided into 4 classes, i.e., Normal, Benign, *In situ* carcinoma, and Invasive carcinoma. The division is equal, having 100 images in each class, making it a balanced dataset (see Fig. 1). According to the data released by conference organizers in [27], the images

were annotated by two medical experts and those images were discarded where the two experts had any disagreements. The images are originally provided in *.tiff* format and have three channels (RGB). The organizers of the BACH 2018 challenge have also provided patient details in separate files for both microscopy and WSI data. They have labelled the patient ids from 1 till 39. The WSI images were extracted from patient 1 to 10 whereas Microscopy images were extracted from patient 11 to 39. The excel files containing data of patient ids can be viewed and downloaded from the URL <https://iciar2018-challenge.grand-challenge.org/Dataset/>. Therefore, the two datasets are extracted from different patients and hence different samples from unique patients helped to validate the performance of the proposed method. More detailed information about the microscopy dataset was provided in their article [27] and the challenge website [21]. Since the organizers did not release the test dataset labels, hence we did not consider the test dataset (100 images) in our work.

The second dataset of Whole-slide images are high-resolution images of digitized sampled biopsy tissues. Each WSI contains more than one pathological labels (Benign, *In situ* carcinoma, and Invasive carcinoma). All the unannotated regions are considered as normal. The challenge provided only 10 annotated WSIs and 20 unannotated WSIs for training. For testing, 10 more slides were released but, without labels. So, we had only 10 training WSI for both testing and training purposes. From each WSI, after the region extraction step elaborated in Section 3.2.1, the distribution of labels is shown in Table 2. The regions have different dimensions and were extracted at the highest resolution level. The understanding of resolution levels of WSI can be understood from Figure 5. Out of the 109 Invasive regions originally annotated by the pathologists, seven regions could not be read by the available computing resource due to their high dimension and memory constraints. Hence, we processed 102 Invasive regions in our work. The images were digitized in *.svs* format and could only be accessed with ASAP or similar software. The organizers also provided the python code to read the annotation files.

4.2 Dataset Preparation

The dataset obtained from the challenge had to be pre-processed for them to be feasibly used for input in the network. The high dimensional WSI regions for the purpose were broken into patches of size 256×256 . The process is explained in Section 3.2.2. The total accumulated patches from WSI regions were 16,934 for the three classes. The same process for patch extraction

Table 2: Distribution of the labels for the microscopy and WSI datasets

Dataset	Benign	<i>In situ</i>	Invasive	Normal
Microscopy	100	100	100	100
WSI	57	109	60	-

- denotes no annotated normal regions

has been repeated for microscopy dataset where each image was of fixed dimension 2048×1536 . For patch extraction step, Microscopy images were divided into a grid of 8×6 dimensions. We call it a grid of patches with each patch of 256×256 dimensions and total 48 patches were acquired from each histology microscopy image. From this dataset, total 19,200 patches were extracted. The accumulated patches from each region were then divided in the form of sets with variable patch numbers.

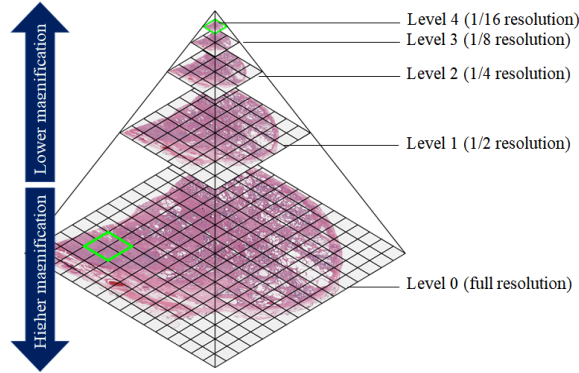


Figure 5: WSI file pyramid structure

4.3 Data Usage

The sequences of features formed after feature extraction process were divided into training, validation and testing sets in the ratio 0.7:0.15:0.15 during parameter selection experiments. After deciding the optimal parameters on the hold-out sets, we followed the 10 fold cross-validation test to verify our results. The distribution of the data during parameter selection experiments is shown in Table 3. All the images extracted from microscopy

Table 3: Distribution of the data for the microscopy and WSI datasets into training, validation and testing sets (for parameter selection only (refer Section 4.4))

Dataset		Benign	Invasive	<i>In situ</i>	Normal
Microscopy	Train	66	75	76	63
	Validation	18	11	12	20
	Test	16	14	12	17
WSI	Train	34	74	45	-
	Validation	11	13	9	-
	Test	12	15	6	-
- denotes no annotated normal regions					

and WSIs had dimensions 256×256 and they were used in their raw form without any color normalization and adjustment.

4.4 Experiments

The pre-trained architecture for extracting deep features was selected through experiments. We tested the performance of end-to-end architecture using ResNet101 and DenseNet201 as feature extractors. The accuracy obtained with ResNet101, *Scan_2*, and WSI dataset was 63.64% whereas with Microscopy dataset, we obtained the accuracy of 84.85%. Similarly, the accuracy obtained with DenseNet201, *Scan_2*, and WSI dataset was 77.97% whereas 71.19% with Microscopy dataset. The choice of *Scan_2* and different hyper-parameters used to train the model with ResNet101 and DenseNet201 was validated in our next experiment.

The second experiment comprise optimal hyper-parameter selection using heuristics and best scanning technique for histopathological images irrespective of the dataset. First, we experimented with three optimizing functions- Stochastic Gradient Descent with Momentum (SGDM), RMSprop, and ADAM. The specific hyper-parameters for each optimizer is summarized as:

SGDM- Momentum: 0.90,

RMSprop- SquaredGradientDecayFactor: 0.9900, Epsilon: $1.0000e - 08$,

ADAM- GradientDecayFactor: 0.9000, SquaredGradientDecayFactor: 0.9900,

Epsilon: $1.0000e - 08$.

These are the default parameter settings in MATLAB2019a and we used them as is. We made the combinations of the chosen hyperparameters which were four dropout rates, three scanning techniques, three optimization functions and two learning rates. The symbol '*' in-front of some of the accuracy values represents that the epochs were run keeping training option of validation patience at 5. The setting ensures that the training stops if the validation loss is larger than or equal to the previously recorded smallest loss for at most 5 times during the training or if the maximum number of epochs are exhausted, whichever is the earlier. In this setting, the number of epochs may or may not reach the maximum limit set at the start of the training. So, we performed all the 72 experiments with and without validation patience 5 for a maximum of 30 epochs. We have shown only the largest of the two accuracy values obtained from the two settings. The '*' indicates that the larger accuracy value is obtained with validation patience 5. So, in total, we conducted $4 \times 3 \times 3 \times 2 \times 2 = 144$ experiments for each dataset to select the optimal hyper-parameters. The experimental results are indicated in table 4 for 3-class classification of WSI tumour regions and table 5 for 4-class classification accuracy of Microscopy dataset. . Several deductions were made from the table 4. Such as, across all the scanning methods, learning rate 10^{-4} performed better than learning rate 10^{-3} . However, for second scanning method (*Scan_2*), both the learning rates performed closely with accuracy values falling in the range 80-88%. Scanning method *Scan_3* followed closely in terms of frequency of accuracy values more than 80%. When we kept the optimization function, learning rate and scanning method constant, the trend of accuracy across different drop-out rates signify the importance of tuning drop-out values during training custom models. With respect to optimization function and irrespective of the drop-out rates, the all-over analysis of the table 4 suggests that SGDM did not perform well in first two scanning methods (*Scan_1* and *Scan_2*) whereas, the gain in SGDM performance was observed in *Scan_3*. In case of ADAM, this optimization function could not enhance model's performance across all hyper-parameters except in *Scan_2* with learning rate 10^{-4} . The optimization function RMSprop performed consistently better across scanning methods *Scan_2* and *Scan_3* irrespective of the learning rates and drop-out rates. The highest performance as can be seen in the table 4 for WSIs was given by *Scan_2*, RMSprop, 0.5 drop-out rate and 10^{-4} learning rate. The cell is highlighted in magenta.

The analysis of table 5 also gives some interesting insights about the behaviour of model when the hyper-parameters change. These values were

Table 4: Accuracy (%) obtained against different learning rates, rates, optimizing functions and scanning techniques with respect to Whole Slide Images (3-classes).

Scanning Method	Learning Rate	Optimizer	Dropout Rate			
			0.4	0.5	0.6	0.7
<i>Scan_1</i>	10^{-4}	SGDM	51.52	57.58	54.55	39.39
		RMSprop	72.73*	60.61*	66.67	63.64
		ADAM	69.70	69.70	66.67	63.64
	10^{-3}	SGDM	72.73*	72.73	66.67*	66.67*
		RMSprop	60.61	66.67*	63.64*	57.58*
		ADAM	51.52	63.64	63.64	63.64
<i>Scan_2</i>	10^{-4}	SGDM	60.61	57.58	57.58	57.58
		RMSprop	75.76	87.88*	81.82*	84.85
		ADAM	78.79	75.76	78.79	84.85
	10^{-3}	SGDM	81.82	84.85	81.82*	81.82
		RMSprop	78.79	75.76*	81.82	69.70
		ADAM	66.67	72.73	72.73	66.67
<i>Scan_3</i>	10^{-4}	SGDM	72.73	75.76	66.67	69.70
		RMSprop	78.79*	72.73	84.85*	78.79
		ADAM	81.82*	69.70*	75.76	72.73*
	10^{-3}	SGDM	78.79	75.76*	78.79*	72.73*
		RMSprop	75.76	78.79	72.73	75.76*
		ADAM	69.70	69.70	66.67	75.76

* validation patience 5

obtained after the 4-class classification of microscopy dataset. The parameters are most sensitive to scanning methods in this dataset as we could observe from the table 5 that when the patches extracted from *Scan_3* were trained using the same hyper-parameters, absolute drop in the accuracy was recorded. The results also indicate of the fact that scanning techniques can over-power the outcome of the model especially in the case of sequence modelling of images to labels. In Microscopy dataset as well, the learning rate 10^{-4} performed better than 10^{-3} and the scanning method *Scan_2* gave better outcome in comparison to other two methods. We observed the difference in optimization function (ADAM) and drop-out rate (0.6) when compared with best performing hyper-parameters in WSI dataset. The hyper-parameter tuning gave us the insight as to how our model behaves which helped us to finally chose our parameter set to perform cross-validation. We deduced that learning rate 10^{-4} and scan technique *Scan_2* with validation patience gave us the better results in both the datasets.

The direct analysis of comparative methods in literature [25, 24, 26] with our proposed method could not be achieved since these methods have different objectives like calculating HER2 scores, five year disease specific survival prediction, hazard ratios. Also, they have different data values associated with each image to facilitate survival analysis on their datasets. Whereas, we do not have such type of data and hence the objectives are different. However, all these methods used CNN + LSTM as their backbone model. Therefore, for indirect qualitative analysis, we performed experiments with one LSTM layer instead of BiLSTM layer while keeping all the other hyperparameters unchanged. For microscopy dataset, we achieved the 10 fold cross-validation overall accuracy of 88.75%. Whereas, we achieved overall accuracy of 54.55% for WSI dataset. Table 6 records the classwise results with LSTM layer. From the obtained results, we observed that the performance with LSTM layer has degraded in comparison to BiLSTM layer. Moreover, with WSI dataset, the performance degradation is quite significant relative to what we observe with Microscopy dataset. Therefore, besides philosophical justification, the short experimental observation also strengthened the choice of BiLSTM over LSTM layer.

4.5 Results and comparison with current literature

The experiments on hyper-parameters tuning on both datasets gave us the optimal set to cross-validate the final accuracy value obtained over the two datasets. For benchmarking purpose we evaluated the performance of state of the art deep learning models- ResNet50[28], InceptionV3[29], and

Table 5: Accuracy (%) obtained against different learning rates, drop-out rates, optimizing function and scanning techniques with respect to Microscopy Images (4-classes).

Scanning Method	Learning Rate	Optimizer	Drop-out Rate			
			0.4	0.5	0.6	0.7
<i>Scan_1</i>	10^{-4}	SGDM	59.32*	59.32	55.93	62.71
		RMSprop	74.58	72.88	74.58	69.49
		ADAM	76.27	72.88	79.66*	76.27
	10^{-3}	SGDM	69.49	69.49	69.49	67.80
		RMSprop	67.80	69.49*	64.41*	62.71
		ADAM	59.32*	69.49*	59.32	71.19*
<i>Scan_2</i>	10^{-4}	SGDM	55.93	71.19*	72.88*	71.19*
		RMSprop	79.66	83.05*	77.97*	81.36
		ADAM	76.27	81.36	84.75*	76.27*
	10^{-3}	SGDM	76.27	81.36*	83.05*	83.05*
		RMSprop	62.71	72.88*	79.66*	74.58*
		ADAM	71.19*	71.19*	81.36*	74.58*
<i>Scan_3</i>	10^{-4}	SGDM	18.69*	22.03*	22.03*	23.73*
		RMSprop	1.69*	1.69*	1.69*	1.69*
		ADAM	0	3.39*	0	3.39*
	10^{-3}	SGDM	3.39*	5.08*	3.39*	5.08*
		RMSprop	10.17*	15.25*	11.86*	16.95*
		ADAM	8.47	15.25*	13.56*	22.03*

* validation patience 5

Table 6: Experimental results for proposed model with LSTM layer.

Dataset	Acc	Benign		Invasive		<i>In situ</i>		<i>Normal</i>	
		Se.	Sp.	Se.	Sp.	Se.	Sp.	Se.	Sp.
Microscopy	0.8875±0.0056	0.8516±0.0106	0.9353±0.0050	0.9303±0.0103	0.9653±0.0061	0.8644±0.0081	0.9416±0.0039	0.9789±0.0045	0.9933±0.0014
WSI	0.5455±0.0629	0.4917±0.0410	0.8967±0.0116	0.7277±0.0115	0.5885±0.0421	0.7508±0.0481	0.8505±0.0376	-	-

DenseNet201[30]. We used the constant learning rate of 10^{-4} and SGDM as optimizer for fine-tuning each of these networks. The last fully connected layer of each of these models were removed and replaced with our new fully connected layer having four outputs in the case of Microscopy dataset and three outputs for the WSI dataset. Each model was trained for 30 epochs. After the model training, we performed majority voting scheme to predict the final label for the image. This process was done for both Microscopy and WSI dataset. The benchmark models are not end-to-end due to the required post-processing of patch-based classifier outputs for image label prediction.

We compared the results from benchmark models and the results by top 5 teams in BACH grand challenge [21] published in [27] with our proposed method on the Microscopy dataset in table 7. Similarly, for WSI dataset, we compared our model’s performance in table 8. We performed 10 fold cross-validation on our proposed model. We have evaluated the performance of our model in terms of overall accuracy of the model, class-wise sensitivity, and specificity.

Sensitivity and Specificity are commonly used for measuring medical applications. Sensitivity refers to how much our model is sensitive in detecting positive class or the percentage of actual positives that are correctly identified. Whereas, Specificity is the measure of actual negatives that are correctly identified. Both Sensitivity and Specificity of the model should be as high as possible to be able to correctly detect all positive samples and all negative samples.

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (3)$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive} \quad (4)$$

4.5.1 Performance on Microscopy dataset

The accuracy of our model is 3% more than the top performing team 216 *Chennaswamy et al.*. The authors also used pre-trained CNNs instead of building their own custom model. They used ensemble of ResNet-101 [28] and two DenseNet-161 [30] networks. In comparison to our model which is end-to-end, they first trained ResNet-101 and a DenseNet-161 using images normalized with breast-histology data and then another DenseNet-161 with images fine-tuned with ImageNet normalization. During the testing, the majority voting scheme was used to declare the class of the input image

Table 7: Comparative Performance metrics with standard errors for patch-to-image classification model for Microscopy Dataset (4-classes)

Method	Acc	Benign		Invasive		<i>In situ</i>		Normal	
		Se.	Sp.	Se.	Sp.	Se.	Sp.	Se.	Sp.
Ours (proposed)	0.9000 \pm 0.0053	0.9400 \pm 0.0052	0.9467 \pm 0.0059	0.9800 \pm 0.0042	0.9800 \pm 0.0023	0.9600 \pm 0.0052	0.9500 \pm 0.0039	0.7200 \pm 0.0235	0.9500 \pm 0.0022
Chennaswamy et al., 2018 (team 216)* [27]	0.87	0.8	0.96	0.88	0.99	0.84	1.0	0.96	0.88
Kwok et al., 2018 (team 248)* [27]	0.87	0.72	0.96	0.92	0.96	0.88	0.97	0.96	0.93
Brancati et al., 2018 (team 1)* [27]	0.86	0.68	0.97	0.96	0.95	0.84	0.99	0.96	0.91
Wang et al., 2018 (team 157)* [27]	0.83	0.64	0.99	0.8	0.97	0.92	0.91	0.96	0.91
Kone et al., 2018 (team 19)* [27]	0.81	0.4	0.99	0.92	0.89	0.92	0.92	1.0	0.95
Roy et al., 2019 * [6]	0.90	0.70	1.0	1.0	1.0	1.0	0.93	0.90	0.93
ResNet50 [28]	0.8675 \pm 0.0083	0.8674 \pm 0.0239	0.9341 \pm 0.0052	0.9243 \pm 0.0121	0.9666 \pm 0.0048	0.8307 \pm 0.0090	0.9706 \pm 0.0047	0.8605 \pm 0.0113	0.9542 \pm 0.0103
DenseNet201 [30]	0.8900 \pm 0.0107	0.8624 \pm 0.0227	0.9697 \pm 0.0039	0.8408 \pm 0.0209	0.9836 \pm 0.0029	0.9170 \pm 0.0099	0.9440 \pm 0.0055	0.9416 \pm 0.0080	0.9575 \pm 0.0093
InceptionV3 [31]	0.8700 \pm 0.0066	0.8499 \pm 0.0163	0.9463 \pm 0.0049	0.8628 \pm 0.0116	0.9735 \pm 0.0045	0.8557 \pm 0.0122	0.9599 \pm 0.0035	0.9125 \pm 0.0145	0.9476 \pm 0.0094

*The standard error data for comparative literature is not available

from among the three classes predicted by the three models. Other notable difference between our model and theirs is that they used bilinear interpolation to resize their image dimensions from 2048×1536 to 224×224 whereas, we did not use resized images since that would have decreased the quality of extracted features. We maintained the resolution and instead broke the image into patches to decrease the size of the input image. At the training time, for feature extraction step through GoogleNet, the patches were resized from 256×256 to 224×224 . Second team on the leaderboard *Kwok et al.* team 248 trained their model using images from both microscopy dataset and extracted patches from WSI dataset. Their 2-stage process first trained the ResNet-v2 [29] pre-trained on ImageNet on patches acquired from microscopy dataset and then again pre-trained their network with the patches acquired from WSI dataset. The prediction of each patch was then aggregated to image-wise prediction. Their method was also not end-to-end and required two datasets to fine-tune the model performance. The difference between accuracy between our model and theirs was also 3%. The class-wise comparison (Table 7) suggests that our model is much sensitive then the top 2 performing teams. Team 1 *Brancati et al.* also used the ensemble of three ResNet models having 34, 50, and 101 layers, respectively. They used down-sampled microscopy images to extract patches of two sizes 308×308 and

615×615 . These patches were taken from the center of the down-sampled images. They used highest class probability from three models as the class of the image. Our model performed better than theirs by overall accuracy of 90% against 86%. The next team in the list was Team 157 *Wang et al.*. The authors in this work trained VGG16 [32] using sample pairing data augmentation technique by [33] in which samples from different classes are augmented and then merged. The merged images are then trained using the chosen model. In the next step, the trained classifier from the mixed images is again trained using the initial non mixed dataset. The authors have resized their images to 256×256 and then extracted patches of size 224×224 at random locations. They achieved the accuracy of 83%. The difference between their and our approach is same as with other competitive models. Team 19 *Kone et al.* achieved the accuracy of 81%, 9 percent less than our proposed model. They proposed binary tree like structure of 3 ResNeXt50 [34] models in which the top CNN in the hierarchy classifies images into carcinoma (*In situ*, Invasive) and non-carcinoma normal and benign. The next two children of the root CNN then classifies the images into respective two sub-classes benign or normal and, *In situ* or Invasive. They also used the two-stage process that used the learned weights of first stage to train the subsequent stages. All these methods in the challenge [21] who have reported their models performance used current state of the art deep learning models. The common thread between these models was that all used pre-trained models due to limited amount of data. However, they all used very heavy resizing of images which compromise with the quality of the high resolution intrinsic details present in cancer data. Moreover, their methods used two-three stages of training and the final outputs were aggregated to declare imagewise prediction. Our model on the other hand as mentioned avoid the disadvantages posed by the compared models. The same disadvantages are posed by the authors in [6] as well. They extracted different size patches (64×64 , 128×128 , 512×512) to train their model separately but found optimum performance with 512×512 . They then used heavy data augmentation to increase the amount of data. The augmented dataset is then trained using their custom CNN architecture. After the patches were trained, they used majority voting scheme to declare the predicted class of the input image. Although, they have achieved equal accuracy as our proposed model but suffered from the drawback of stage-wise model, data augmentation, and having to train their model from scratch which demands time and space.

Table 8: Comparative Performance metrics with standard errors for patch-to-image classification model for WSI Dataset (3-classes)

Method	Acc	Benign		Invasive		<i>In situ</i>	
		Se.	Sp.	Se.	Sp.	Se.	Sp.
Ours (proposed)	0.8402±0.0032	0.7090±0.0309	0.9132±0.0157	0.9142±0.0136	0.9190±0.0096	0.8333±0.0264	0.9240±0.0117
ResNet50 [28]	0.8127±0.0093	0.9233±0.0202	0.8341±0.0148	0.8285±0.0113	0.9492±0.0126	0.7167±0.0271	0.9556±0.0065
DenseNet201 [30]	0.8127±0.0054	0.8142±0.0202	0.9091±0.0071	0.8520±0.0098	0.9160±0.0135	0.7833±0.0500	0.9056±0.0079
InceptionV3 [31]	0.8221±0.0087	0.8356±0.0140	0.8740±0.0142	0.8451±0.0077	0.9183±0.0087	0.7667±0.0245	0.9369±0.0696

4.5.2 Performance on WSI dataset

Microscopy dataset has balanced sets of four classes having equal image dimensions. The labelled mask of each class covers the entire image area and hence the features detected belong to one class only. These properties have helped to capture patches that completely belong to the labelled image class. however, with WSI dataset, due to arbitrary shape and size of the regions, the automatic extraction script could only extract the tumour from the surrounding bounding box area. Hence, the patches sampled from such WSI regions also contained a lot of non-tumour or non-class images. Moreover, the final acquired image regions were imbalanced (Table 3). Therefore, these reasons might have caused the performance decline in the accuracy with WSI dataset in comparison to microscopy images. We trained for only three classes since the normal patches were randomly extracted and therefore, did not belong to one particular area in the WSI. The continuity of the patches is the important factor for our model. For experimental purposes when we trained our model with non-continuous normal patches, our model suffered from performance decline which proved that the continuous patches draw spatial and contextual relationship through BiLSTMs. Otherwise, in the absence of non-continuity, the model may suffer from high variance. For benchmarking purposes and due to the lack of other comparative models, we compared our model with ResNet50, InceptionV3, And DenseNet121. From the Table 8, we could observe an improvement in the performance metrics when we used context based model. The main difference between our model and these state of the art models is that we did not train any deep architecture and our model is end-to-end.

5 Discussions

Computer Aided Diagnosis (CAD) by analysing samples of Ultrasound, CT, and MRI images has been vastly suggested by medical image researchers for quite sometime. They trained machine learning models with various morphological, graph, and intensity based methods from very small set of data samples which were sometimes in the range of only 30 to 100 images. The generalizing capability of such models has thus been questionable. However, after the introduction of deep learning models and availability of large amount of data. CAD techniques have experienced a huge success in performance precision and accuracy. When such deep models were tested for histopathological images, the low inter-class variability, especially between Normal and Benign classes, affected the overall performance. Hence, some new methods engaging these deep models in form of cascaded or ensemble architectures were proposed. Also, the most biopsy samples digitized at high resolutions contain very detailed information of cell structures and various other microstructures. The amount of information in one biopsy sample could collectively form a gigapixel image. Such high-resolution images are then required to be broken into smaller patches for further processing. Patch-based processing with complex ensemble methods followed by aggregation of patches into image labels in case of classification and segmented objects in case of segmentation makes it a lengthy process. The whole pipeline is divided into stages and lacks contextual relationship between patches. To overcome this drawback, we thought to streamline the process into an end-to-end network. The patches were visualized as a sequence of images as in a video and an effort was made to scan the patches so as to maintain as much continuity as possible. RNN based BiLSTM models are known to serve the purpose for predicting input sequence labels. Since, with BiLSTMs., we could capture both past and future contexts which enabled the model to aggregate the whole tumour features despite providing non-overlapping tumour parts in the form of patches as input sequence.

Due to sequence classification, the next step of predicting image label from patch labels was not required. The graphical structure of BiLSTMs helped to build a context-based high-resolution tumour classification model that also gave us the benefit of end-to-end network structure. We also analysed that with our proposed models, there is no need of training deep models. We used a pre-trained ImageNet model for feature extraction and only one BiLSTM layer to train a shallow network. The average time to train the model was 17 minutes for 30 epochs. Once the model's hyper-parameters are tuned for the particular dataset, the training would take only few minutes.

The shallow structure of the model also make it feasible for deployment in lighter applications such as hand-held devices like mobile handsets. The complexity of the method is discussed in section 6. Another advantage is that the various limitation of high-resolution images could be exploited in the favour of the methodology. The large dimensions could be easily turned into sequences using the appropriate scanning process. Due to BiLSTM layer, the model encapsulates the context mining capability which helped form the spatial and contextual relationship between patches sampled from a single image. The results suggested that this context modelling was crucial in patch-based models that process patches instead of complete structures at a time. In other words, modelling direct dependencies between patches, past or future, is crucial for performance of the model.

The idea of processing patches as a sequence using RNN based BiLSTM model could be further extended by using four RNNs. Each RNN would take patches going in up, down, left, and right directions, respectively [35]. According to the [36, 37], compared to CNNs, RNNs when applied to images allow for long-range lateral interactions between features in the same feature map.

6 Complexity

The model is end-to-end deep learning model whose architecture is briefly expressed in Table 1. Till layer number 143 – *FlattenLayer*, there are no Floating Point Operations (FLOPs) being performed. GoogLeNet network is present to extract pre-trained features which are then passed on to subsequent layers for further processing. Similarly, Sequence Folding, Unfolding, Average Pooling, and Flatten layer also accumulate zero FLOPs. Therefore, the time complexity is calculated from BiLSTM layer onwards. The formula for calculating number of learnable parameters in a BiLSTM layer is derived as follows,

Let I be the input size of the sequence, K be the number of output dimensions and H be the number of hidden units. For BiLSTM if H are the number of initialized hidden units then $M = 2 \times H$ are the total number of hidden units for both forward and backward passes of the BiLSTM network. After concatenation of the forward and backward outputs, the total output dimensions become $K = M/2$. Then the complexity of a BiLSTM layer is:

$$\mathcal{O}(W)$$

where W are the total number of learnable parameters in the network calculated as:

$$W = 4 \times M((I + 1) + K)$$

$$W = 4 \times (M(I + 1) + MK)$$

Here in the above formula, the first term $4 \times M(I + 1)$ are the total number of input weights and the second term $4 \times MK$ are the number of recurrent weights.

In the terms of Big Oh notation, the time complexity is;

$$\mathcal{O}(M(I + 1) + MK)$$

The multiplication by factor 4 represents four weight matrices of BiLSTM layer (Input gate, Forget gate, Cell candidate, Output gate). The input size variable I is added with a bias value 1.

For the BiLSTM layer in our network, the number of parameters are:

$$W = 4 \times 4000 \times ((1024 + 1) + 2000)$$

$$W = 16000 \times (1025 + 2000)$$

$$W = 48400000$$

where 4000 are the total number of hidden units for both forward and backward passes of the BiLSTM layer, 1024 is the size of the input sequence, and 2000 is the total number of outputs.

Next, for the fully connected layer, the parameters are

$$F = 3 \times 4000$$

Hence, the total number of FLOPs are

$$W + F = 48400000 + 12000 = 48412000$$

$$W + F = 48.4 \times 10^6 = 48MFLOPs$$

We have used NVIDIA TitanX GPU (12GB) for training our models. It performs 11×10^{12} or 11 Tera FLOPs per second which is a sufficient computational efficiency required for training.

To put it in perspective, we mention the number of FLOPs for few popular deep learning networks in Table 9.

In terms of the Big Oh notation, the time complexity of the model for t number of input samples and n number of epochs is represented as;

$$\mathcal{O}(n \times t \times (W + F))$$

Table 9: FLOPs for popular deep learning architectures

AlexNet	727 MFLOPs
VGG16	16 GFLOPs
VGG19	20 GFLOPs
GoogLeNeT	2 GFLOPs
ResNet50	4 GFLOPs
DenseNet121	3 GFLOPs
InceptionV3	6 GFLOPs

7 Conclusion

We proposed an end-to-end RNN based model that takes patches as input and outputs image labels. The patches are modelled as sequences by using one-layer BiLSTM model. The sequence in an image is captured using the strategic scanning method which was experimentally chosen. We used BACH challenge dataset to test our method and reported our results on two different datasets introduced in the challenge. The classifier performance was compared with recently reported metrics by top 5 teams in BACH challenge for microscopy dataset. We achieved highest performance of 90% with simpler architecture and less time and space complexity.

Acknowledgement

This research was carried out in Indian Institute of Information Technology, Allahabad and supported, in part, by the Ministry of Human Resource and Development, Government of India and the Biomedical Research Council of the Agency for Science, Technology, and Research, Singapore. We are also grateful to the NVIDIA corporation for supporting our research in this area by granting us TitanX (PASCAL) GPU.

References

- [1] A. S.-Y. Leong, Z. Zhuang, The changing role of pathology in breast cancer diagnosis and treatment, *Pathobiology* 78 (2) (2011) 99–114.

- [2] L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, J. H. Saltz, Patch-based convolutional neural network for whole slide tissue image classification, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2424–2433.
- [3] K. Nazeri, A. Aminpour, M. Ebrahimi, Two-stage convolutional neural network for breast cancer histology image classification, in: *International Conference Image Analysis and Recognition*, Springer, 2018, pp. 717–726.
- [4] A. Mahbod, I. Ellinger, R. Ecker, Ö. Smedby, C. Wang, Breast cancer histological image classification using fine-tuned deep network fusion, in: *International Conference Image Analysis and Recognition*, Springer, 2018, pp. 754–762.
- [5] S. Wang, Y. Zhu, L. Yu, H. Chen, H. Lin, X. Wan, X. Fan, P.-A. Heng, Rmdl: Recalibrated multi-instance deep learning for whole slide gastric image classification, *Medical image analysis* 58 (2019) 101549.
- [6] K. Roy, D. Banik, D. Bhattacharjee, M. Nasipuri, Patch-based system for classification of breast histology images using deep learning, *Computerized Medical Imaging and Graphics* 71 (2019) 90–103.
- [7] Y. Huang, A. C.-s. Chung, Improving high resolution histology image classification with deep spatial fusion network, in: *Computational Pathology and Ophthalmic Medical Image Analysis*, Springer, 2018, pp. 19–26.
- [8] M. Shaban, R. Awan, M. M. Fraz, A. Azam, D. Snead, N. M. Rajpoot, Context-aware convolutional neural network for grading of colorectal cancer histology images, *arXiv preprint arXiv:1907.09478* (2019).
- [9] T. Araújo, G. Aresta, E. Castro, J. Rouco, P. Aguiar, C. Eloy, A. Polónia, A. Campilho, Classification of breast cancer histology images using convolutional neural networks, *PloS one* 12 (6) (2017) e0177544.
- [10] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation* 9 (8) (1997) 1735–1780.
- [11] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing* 45 (11) (1997) 2673–2681.

- [12] A. Graves, S. Fernández, J. Schmidhuber, Bidirectional lstm networks for improved phoneme classification and recognition, in: International Conference on Artificial Neural Networks, Springer, 2005, pp. 799–804.
- [13] J. Johnson, A. Karpathy, L. Fei-Fei, Densecap: Fully convolutional localization networks for dense captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4565–4574.
- [14] A. Karpathy, L. Fei-Fei, Deep visual-semantic alignments for generating image descriptions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3128–3137.
- [15] O. Vinyals, A. Toshev, S. Bengio, D. Erhan, Show and tell: A neural image caption generator, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3156–3164.
- [16] J. Zhang, Q. Wu, C. Shen, J. Zhang, J. Lu, Multilabel image classification with regional latent semantic dependencies, *IEEE Transactions on Multimedia* 20 (10) (2018) 2801–2813.
- [17] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, W. Xu, Cnn-rnn: A unified framework for multi-label image classification, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2285–2294.
- [18] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, Cnn: Single-label to multi-label, *arXiv preprint arXiv:1406.5726* (2014).
- [19] Y. Guo, Y. Liu, E. M. Bakker, Y. Guo, M. S. Lew, Cnn-rnn: a large-scale hierarchical image classification framework, *Multimedia Tools and Applications* 77 (8) (2018) 10251–10271.
- [20] B. 2015, 4th international symposium in applied bioimaging, <http://www.bioimaging2015.inib.up.pt/> (2015).
- [21] I. B. 2018, Grand challenge on breast cancer histology, <https://iciar2018-challenge.grand-challenge.org/Home/> (November 2018).
- [22] F. A. Spanhol, L. S. Oliveira, C. Petitjean, L. Heutte, Breast cancer histopathological image classification using convolutional neural networks, in: 2016 international joint conference on neural networks (IJCNN), IEEE, 2016, pp. 2560–2567.

- [23] N. Bayramoglu, J. Kannala, J. Heikkilä, Deep learning for magnification independent breast cancer histopathology image classification, in: 2016 23rd International conference on pattern recognition (ICPR), IEEE, 2016, pp. 2440–2445.
- [24] T. Qaiser, N. M. Rajpoot, Learning where to see: A novel attention model for automated immunohistochemical scoring, *IEEE transactions on medical imaging* 38 (11) (2019) 2620–2631.
- [25] J. Ren, K. Karagoz, M. Gatza, D. J. Foran, X. Qi, Differentiation among prostate cancer patients with gleason score of 7 using histopathology whole-slide image and genomic data, in: *Medical Imaging 2018: Imaging Informatics for Healthcare, Research, and Applications*, Vol. 10579, International Society for Optics and Photonics, 2018, p. 1057904.
- [26] D. Bychkov, N. Linder, R. Turkki, S. Nordling, P. E. Kovanen, C. Verrill, M. Walliander, M. Lundin, C. Haglund, J. Lundin, Deep learning based tissue analysis predicts outcome in colorectal cancer, *Scientific reports* 8 (1) (2018) 1–11.
- [27] G. Aresta, T. Araújo, S. Kwok, S. S. Chennamsetty, M. Safwan, V. Alex, B. Marami, M. Prastawa, M. Chan, M. Donovan, et al., Bach: Grand challenge on breast cancer histology images, *Medical image analysis* (2019).
- [28] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [29] C. Szegedy, S. Ioffe, V. Vanhoucke, A. A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in: *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [30] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision. *arxiv* 2015, *arXiv preprint arXiv:1512.00567* 1512 (2015).
- [32] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556* (2014).

- [33] H. Inoue, Data augmentation by pairing samples for images classification, arXiv preprint arXiv:1801.02929 (2018).
- [34] S. Xie, R. Girshick, P. Dollár, Z. Tu, K. He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1492–1500.
- [35] Y. Bengio, I. Goodfellow, A. Courville, Deep learning, Vol. 1, Citeseer, 2017.
- [36] F. Visin, K. Kastner, K. Cho, M. Matteucci, A. Courville, Y. Bengio, Renet: A recurrent neural network based alternative to convolutional networks, arXiv preprint arXiv:1505.00393 (2015).
- [37] N. Kalchbrenner, I. Danihelka, A. Graves, Grid long short-term memory, arXiv preprint arXiv:1507.01526 (2015).