A Thesis

for the Degree of Master of Sciences

Use of Artificial Intelligence for the Identification of Barrier Integrity in Porcine Small Intestinal Epithelial Cells

인공지능에 의한 돼지 소장상피세포의 장벽기능 식별

February 2020

By

Payam Hosseinzadeh Kasani

Department of Agricultural Biotechnology

Graduate School, Seoul National University

농학석사학위논문

Use of Artificial Intelligence for Barrier Integrity Detection in Porcine Small Intestinal Epithelial Cells

인공지능 활용 돼지장상피세포의 장벽 결함 감지

Under the guidance of Professor Cheol-Heui Yun

Submitting a master's thesis of Agricultural Biotechnology

February 2020

Graduate School of Agricultural Biotechnology

Seoul National University

Payam Hosseinzadeh Kasani

February 2020

This thesis has been evaluated in detail by each member of the committee member of the following supervisory committee and has been found satisfactory

Chair of committee	Byung-Chul Park	(Seal/signature)
Vice Chair of committee	Cheol-Heui Yun	(Seal/signature)
Member of committee	Tea-Sub Park	(Seal/signature)

Summary

Given the importance of monitoring intestinal permeability and the significant healthcare cost associated with the gut barrier disruption, automatic detection of barrier disruption pattern in porcine intestinal epithelial cells (IPEC-J2) using automatically computer-aided detection models based on a deep convolutional neural network for early detection and interpretation in measuring intestinal permeability is an area of active research and adequate experimental models are required to further understand the grade of localization and disruption of IPEC-J2 tight junction proteins.

In the present study, a deep learning-based ensemble model to build a classifier to automatically analyze and extract features from input images in order to accurately assess the grade of localization and disruption of tight junction proteins (TJ) in IPEC-J2 have been proposed. Different data augmentation techniques including horizontal and vertical flips, rotating, zooming, contrast adjustment and brightness enhancement with different parameters are employed to increase the dataset size and tackle the over-fitting problem. At first, the experiments began with evaluating the performance of 8 state-of-the-art deep CNN architectures namely, VGG-Net, InceptionV3, MobileNet, DenseNet, Xception, NAS-Net, InceptionResNetV2 and ResNet models on IPEC-J2 cell image classification. Transfer learning is a common strategy in training deep CNN models. Using this strategy, the weights that are already learned on a cross-domain dataset to initialize weights of deep CNN models can be transferred in this research. The final results showed that the deep CNN ensemble of InceptionV3 and DenseNet201 achieved the best result with an accurate detection rate of 99.22% than the individual InceptionV3 architecture (95.03%) and the individual DenseNet201 architecture (91.11%). The second-best ensemble architecture was the ensemble of InceptionV3 and MobileNet with an accuracy of 97.78% than the individual InceptionV3 architecture (95.03%) and the individual MobileNet architecture (95.82%.)

Collectively, employing CNN models could be considered as an automatic visual inspection system for the recognition, grading of expression, localization and disruption of tight junction proteins in epithelial cells with less misdiagnosis (false positive or false negative) and error rate, and also reduce the heavy workload of manual diagnosis.

Keywords:

Computer-aided Diagnosis, Convolutional Neural Network, Transfer learning, Intestinal Barrier integrity, Tight junction Proteins, Porcine Intestinal Epithelial Cell

Student number: 2017-21969

Contents

List of Tables

Review of Literature

Experiments

Results

List of Figures

Review of Literature

Materials and Methods

Results

Experiments

I. Review of Literature

1. Convolutional neural network

1.1 Characterization and design of generalized convolutional neural network

Convolutional neural network (CNN or ConvNe), proposed by LeCun et al. [1], is a specialized subtype of multi-layer neural network that designed to recognize visual patterns directly from pixel images. CNN models have achieved promising results in different image recognition, image classification and computer vision tasks [2].

The primary purpose of Convolution is to extract features from the input image [3]. CNNs are made up of a stack layers of neurons that have learnable weights and biases. Layers used to build CNNs will detect a set of features such as lines, edges and contours [4]. In practice, CNN learns the values of these filters on its own during the training process [5]. Technically, deep learning CNN models are used to train and test each input image by transforming the original image feature through layers. CNNs have provided significant breakthrough in many areas including image recognition and classification [3].

CNNs include four main types of layers that are the basic building blocks of every CNN architectures including:

1. Convolution layer
2. Rectified linear unit function
3. Pooling layer
4. Fully connected layer

1.1.1 Convolution layer

A convolutional layer is the core building blocks for designing CNN models [8]. Convolution layer defined by a set filters (kernels). These filters slide over the input image and extract specific features from input images. The essential parameters when designing the convolution layers are as follows:

- The size of filter o kernel window
- The number of filters per layer
- The number of pixels by which filter window stride over the input.
- The padding that defines the border around the filter window. There are two options for padding, valid or same. In valid padding, no padding is defined while in same padding, the padding of zeros is produced around the boundary pixels of the input image so that the size of the convolved image is the same as that of the original image. Therefore, in the same padding, one may not have information lost from the input image.

1.1.2 Rectified linear unit function

Rectified linear unit (ReLU), a commonly used activation function, provides non-linearity to the neural network. The ReLU function is defined by the following formula;

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

ReLU function output the zero or positive value equal to the same value as the input. However, the negative value will be converted to zero [9]. The following is a graph of ReLU (Figure 1);

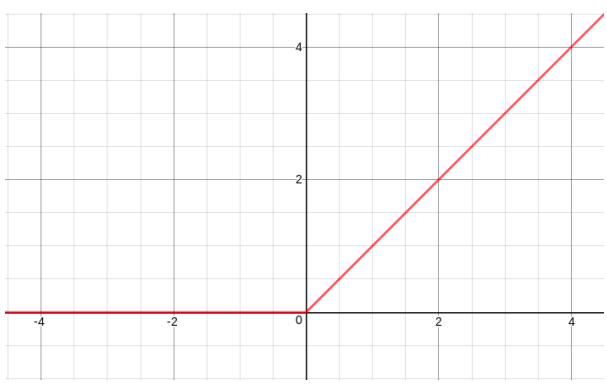


Figure 1. ReLU activation function graph.

1.1.3 Pooling layer

Subsampling is generally used to progressively reduce the spatial size of the input by performing simple mathematical operation such as calculating the average and maximum value of convolution filter and hence speed up the computation but retains the most important information [10]. Pooling layer operates on each feature map independently. Two types of pooling layers exist in designing convolutional neural networks: average pooling and max-pooling [11]. Average pooling involves calculating the average value of the sub-region. Max pooling is a sample-based discretization process and outputs the maximum value of the sub-region. While max pooling selects the brighter pixels from the image and is especially useful when the background of the image is dark and the lighter pixels of the image are interested in average pooling, the sharp features may not be identified when this pooling method is used [12].

1.1.4 Fully connected layer

In fully connected layer, every neuron in the previous layer is connected to every neuron on the next layer. Also, the final output layer in a CNN model is a normal fully-connected layer to produce the final classification decision [13]. Figure 2 shows a typical structure of a CNN Architecture.

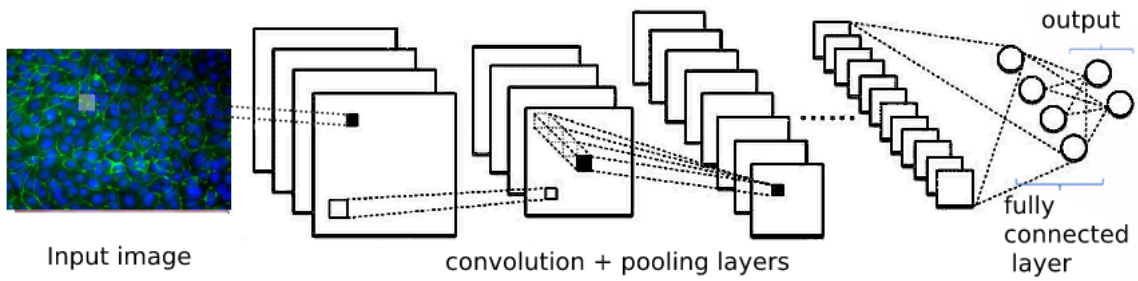


Figure 2. A general structure of a ConNet Architecture.

1.2 Feature extraction using transfer learning

Feature extraction is an essential step in training deep CNN models properly. In this method, instead of training a model from scratch, the weights already trained on a cross-domain dataset transfer into the current classification task. Various studies demonstrated the advantages of transfer learning method in improving the accuracy of the detection rate. Transfer learning strategy has advantages including reducing the computational resources, addressing the overfitting issue due to the insufficient training samples and also speeding up the convergence of the network [14-17]. In this approach, different well-established CNN architectures, InceptionV3, MobileNet, DenseNet, Xception, NAS-Net, InceptionresNetV2 and ResNet, which have shown promising results in many computer vision tasks have been employed [18-20]. For all of these architectures, the initial weights are initialized from the weights learned from ImageNet by transfer learning method. In the following section, a short introduction to the architectures employed in this research is provided.

1.2.1 InceptionV3:

InceptionV3 architecture, proposed by Szegedy et al. in 2014 [22], won the ImageNet competition. This architecture introduced the concept of inception module. The InceptionV3 is the third generation of Inception and consists of 159 layers in total. The main idea of the Inception module is to of different convolution size, i.e. 1×1, 3×3, and 5×5 filter size instead of using one type of kernel. Employing different convolution size allows the multi-level feature extraction from the input image in each convolution operation. Also, by using the pointwise 1×1 convolution, this architecture is able to reduce the number of parameters leading to lessen the computational cost.

1.2.2 Xception:

The Xception architecture, stands for extreme inception, is an extension of the Inception architecture. This architecture introduced by François Chollet in 2017 [23], wherein the Inception modules are replaced with depthwise separable convolutions with residual connections. The depthwise separable convolution can decrease the

memory requirements and computational cost. Xception consists of 36 convolutional layers structured into 14 modules. All modules are connected by linear residual connections, except for the first and last module.

1.2.3 MobileNet:

MobileNet architecture, designed by the Google research team for object recognition on mobile devices [24], consists of depth-wise separable convolution and 1×1 point-wise convolution layers. The performance of the MobileNet architecture is evaluated on ImageNet dataset and achieved an accuracy in the same level of accuracy as VGG16 with 32 times less parameters while is 27 times less computationally intensive [25]. Depth-wise convolution uses a single spatial filter for each input feature map, and pointwise convolution (1×1) applies on cross-channel patterns.

1.2.4 NAS-Net:

NAS-Net (Neural Architecture Search Net) architecture proposed by Google Brain in 2017 [26]. This architecture obtained the state-of-the-art results on the CIFAR10 data set. By employing a recurrent neural network, this architecture can search for the best convolutional layer on the CIFAR-10 dataset and then the selected layers are transferred to the ImageNet dataset. Then the selected suitable convolutional layer stacked together to produce the final model. This architecture introduced a new regularization method called Scheduled Drop Path is to further improve generalization ability. However, in large datasets, the computational cost of this architecture has become expensive.

1.2.5 ResNet50:

ResNet (Deep Residual Learning Network) architecture, proposed by He et al. [27], won the competitions at ILSVRC classification task and achieved excellent results ImageNet and MS-COCO object detection. The concept of a residual block has introduced this architecture to train deeper networks to perform better recognition tasks. The aim of residual blocks to add a connection (instead of concatenation) from the input of the first block to the output of the next block. Residual blocks by using this shortcut connection learn the residual that aid in solving the issues of vanishing gradients and parameter explosion.

1.2.6 DenseNet:

DenseNet (Densely connected convolutional networks) architecture has been proposed [28]. This architecture is an extension for ResNet architecture. In DenseNet architecture, summation operation is used to connect layers to each other. This operation helps to even better generalization ability and handling vanishing gradient issue in compare to ResNet architecture. Feature maps extracted from each layer are reused as input for the next layers. The idea of feature map reusing helps in further improvement of detection rate.

1.2.7 VGG-Net:

VGG-Net (Visual Geometry Group) introduced by Karen Simonyan and Andrew Zisserman in 2014 [29] that achieves top performances on ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This architecture, by using 3×3 filter size provides better features extraction from input images by increasing the depth of the neural network instead of its width. There are two versions of VGG-Net architecture namely, VGG16 and VGG19, with different depths and layers. The performance of both VGG16 and VGG19 is investigated in this research.

1.2.8 InceptionResNetV2:

InceptionResNetV2 employs both Inception and Residual blocks in its architecture[30]. The combination of both modules helps to achieve very good performance with relatively low computational cost.

2. Intestinal barrier and pathways of permeability

The gut barrier is an important component of normal gastrointestinal (GI) tract. The intestinal epithelial layer forms the major barrier that separates our body from the external environment. To prevent an overwhelming immune activation the maintenance of the intestinal epithelial barrier is the essential function of the intestinal epithelial cells (IECs) as it provides a barrier to pathogenic and antigenic components in the lumen that is highly critical for host survival.

Impaired intestinal barrier function has been associated with the pathogenesis of many gastrointestinal diseases, as it facilitates passage of injurious factors such as lipopolysaccharide, peptidoglycan, whole bacteria, and other toxins led to subsequently increased antigen trafficking. Therefore these Intestinal disorders caused a cycle of events resulting in the breakdown of the intestinal gut barrier.

The intestinal barrier includes mechanical, immunological, biological, and chemical barriers. The mechanical barrier is mainly composed of the mucous layer on the surface of the intestinal mucosa, intestinal epithelial cells, intercellular junctions, submucosal lamina propria. The mechanical barrier is a pivotal part of the intestinal barrier and maintained through the intestinal epithelial cells and intercellular junctions [31].

There are four layers in the gut where each layer of barrier has a unique structure and contribution to barrier function. The first, commensal bacteria reside in the gastrointestinal tract which serves as a microbial barrier aids digestion and captures antigen in the lumen. This microbiota is crucial for the maintenance of a symbiotic relationship with immune cells [32]. The second, thin mucous layer contains proteins and defense factors that serve as a chemical barrier and covers the apical surface and limits direct interactions of the epithelium with microbes and larger molecules, such as food particles [33]. The third, intestinal epithelial cells (IECs) form a continuous monolayer and are tightly attached to each other by junctional complexes serves as a physical barrier while supporting nutrient absorption [34]. Finally, the gut associated lymphoid tissue in the lamina propria forms the immunological barrier if antigen penetrates where innate and adaptive immune cells reside such as T cells, B cells, macrophages and dendritic cells [35].

Studies have shown that environmental and psychosocial stressors play a central role in the initiation and (or) exacerbation on GI function and disease susceptibility in humans and animals.

In humans, intestinal barrier defects have been associated with various diseases and disorders, including celiac disease (CD) [36], inflammatory bowel disease (IBD) [37], colon carcinoma, irritable bowel syndrome [38], obesity [39] and food allergies [40]. In agriculturally-important species such as the pig, psychosocial and environmental stressors associated with production practices (e.g., weaning, mixing/crowding stress and heat stress) have significant deleterious impacts on growth performance and GI function [41] and increased susceptibility to infectious GI diseases [42].

Weaning as an early life stressor impairs the intestinal barrier integrity leading to long-lasting deleterious consequences on gut health and susceptibility of diseases in piglets throughout the production lifespan compared with age-matched, non-weaned littermate pigs [43]. Several stressors such as mixing stress, maternal separation, diet change and transportation are imposed upon the pig during weaning which collectively contributes to compromised gut health and well-being of the weaned pig. The impaired intestinal barrier in pigs appears to be linked to immature development of the enteric nervous system and may contribute to the pathogenesis of functional GI disorders both in early life as well as adulthood [44]. Decreased intestinal integrity leads to higher circulating concentrations of endotoxin and altered blood metabolite and inflammation profiles during heat stress in pig [45].

Schematic representation of the main components of the Intestinal epithelial barrier structure and the major tight junction proteins is depicted in Figure 3. The intestinal porcine epithelial cell line J2 (IPEC-J2), as a non-transformed intestinal cell line, originally are intestinal porcine enterocytes derived from the jejunal epithelium of a neonatal unsuckled piglet. Because of the similarity between pig and human intestinal function, studies with IPEC-J2 not only can give more insight when investigating domestic animal study but also provide a valuable tool to imitate the human physiology more closely than any other cell line of non-human origin. Therefore, it is a highly valuable tool in order to study epithelial barrier integrity on

a variety of widely used parameters such as permeability, transepithelial electrical resistance (TEER), a metabolic activity that reflecting epithelial functionality.

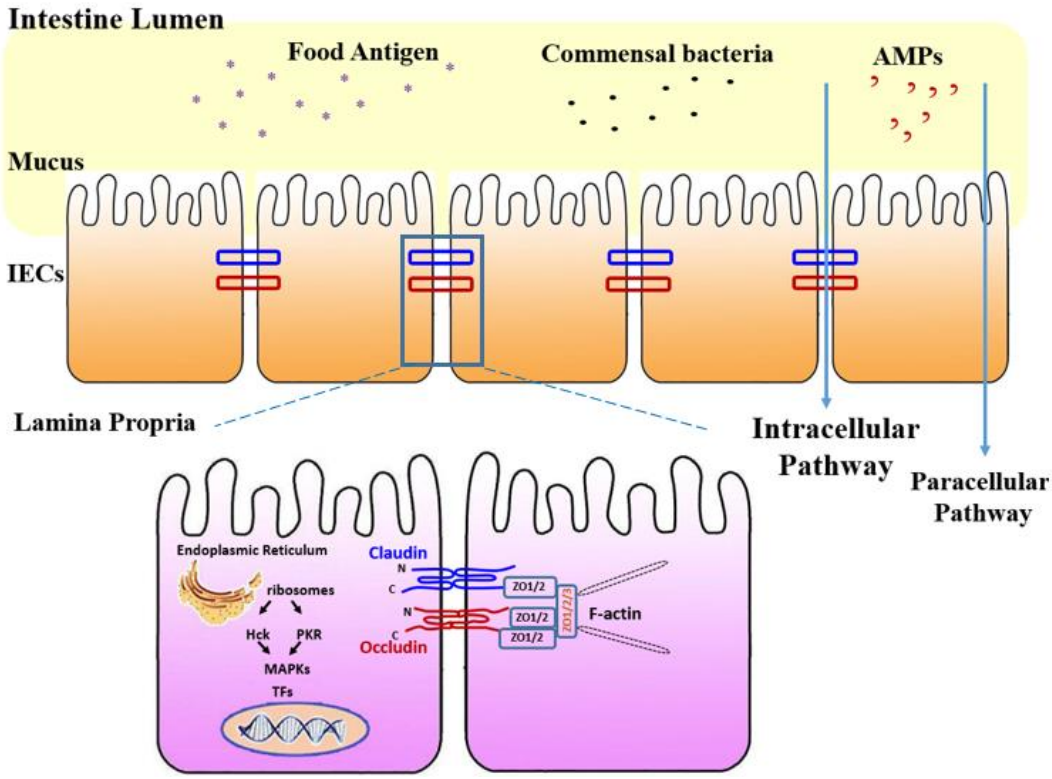


Figure 3. Schematic representation of intestinal epithelial barrier structure and the major tight junction proteins. AMPs: Antimicrobial peptides, IECs: intestinal epithelial cells.

3. Tight junction proteins

3.1 Characterization of intestinal tight junction proteins

The intestinal mucosal surfaces are under continuous contact to trillions of commensal microorganisms and antigens from ingested food products [46]. Mucosal surfaces are lined by a single layer of epithelial cells that separates the intestinal lumen from the underlying lamina propria and act as the first line of defense against the pathogen, physical disruption and harmful environmental factors [47]. These protective barriers are regarded as tight junction's complex [48].

Tight junctions (TJs) are intercellular adhesion complexes that play an essential role in maintaining the intestinal barrier and health of animals [49]. In addition to preventing the entrance of foreign antigens freely across the epithelium, such as pathogens and toxins coming from the intestinal lumen, the intestinal barrier regulates selective permeability for the absorption of nutrients [50]. Moreover, the TJ barrier is actively involved in the regulation of intercellular communication and paracellular transport [49]. The tight junction proteins consisting of occludin, claudins, zonula occludens (ZOs: ZO-1, ZO-2, ZO-3, MUPP-1), which interact with each other, as well as with the cytoskeleton to form a complex architecture [51]. Loss of TJs as a result of either inflammation or other stressor factors impair the proper cell to cell functioning and is thought to induce the pathogenesis of several diseases, which consequently results in increased permeability and activation of mucosal responses [52].

3.2 Zonula occluden family

Zonula occludens (ZO) proteins are scaffolding proteins belongs to a family of multi-domain proteins known as the membrane-associated guanylate kinase (MAGUKs) homologs [53]. There are three different ZO-1 proteins with common structural features, Src homology 3 (SH3) domain, guanylate kinase (GUK) domain and N-terminal region with 3PDZ domains [54]. ZO appears to be a framework to organize the proteins connecting transmembrane proteins to other cytoplasmic proteins and to

actin microfilaments. Moreover, ZO proteins form the central network for protein interactions and are required for assembly of both adherents and tight junctions, thereby contributing to the so-called “junctional plaque” [55]. ZO contains multiple domains that bind a diverse set of junction proteins where the first PDZ domain of all ZO proteins directly interacts with the carboxyl-terminal domain of the claudins, and this association has been attributed to have a central role in TJ assembly and function. Therefore, ZO has been indicating the importance in the regulation of tight junction protein assembly

3.3 Occludin family

The name of occludin derives from the Latin word ‘‘occludere’’, which means to occlude is the first identified transmembrane TJ protein with a 65-kDa located at the TJ [56]. This integral protein of TJ complex is a tetraspan protein comprises of four transmembrane domains, two extracellular loops of similar size, an NH2- and COOH-terminal cytoplasmic domains [57]. Occludin is capable of binding directly through carboxyl segment, to the MAGUK proteins ZO-1, ZO-2 and ZO-3 [58]. Occludin interactions with ZO-1 and this binding are necessary for tight junction barrier regulation.

It has been proposed that occludin interact, directly or indirectly, with claudin [59]. Although occludin is a clear component of TJ complex, its precise role in TJ remains unclear. It has also been revealed that occludin downregulation in intestinal led to an increase in intestinal epithelial TJ permeability [60]. However, the occludin depletion in intestinal epithelial cells, both in vitro and in vivo results in a molecular size-dependent increase macromolecule permeability in paracellular flux [61].

3.4 Claudin family

Mammalian claudins family are 20 to 27 kDa proteins [62] and have four transmembrane domain proteins with two extracellular loops (ECL1–2), cytoplasmic terminal residues, and an intracellular loop (ICL) mediating cell to cell contacts [63].

It is proposed that the first extracellular loop (ECL1) is important for determining the paracellular tightness and the selective paracellular ion permeability, and the second extracellular loop is involved in cell-cell adhesion [64]. The carboxy-termini of claudins contain a PDZ-binding motif and bind to PDZ domains within the cytoplasmic scaffolding proteins ZO-1 and ZO-2, these interactions are thought to be important for junction assembly [65]. Loss of the paracellular barrier-forming claudin leads to inflammation in several mouse knock-out models [66].

Based on their sequence similarity, TJs are divided into classic and non-classic claudins [67]. Some of TJ proteins have sealing functions (claudins 1, 3, 5, 11, 14, 19, and tricellulin). In contrast, a significant number of claudins form channels across TJs which feature selectivity for cations (claudins 2, 10b, and 15), anions (claudin-10a and -17), or are permeable to water (claudin-2).

4. Experimental evaluation of intestinal barrier function

Intestinal permeability and consequences of impaired intestinal barrier integrity can be evaluated with a number of permeability assays along with various marker molecules, alone or in combination, depends on the research question each providing very different information as to treatment endpoints in clinical studies [68]. When utilized and interpreted appropriately, intestinal permeability tests can give researchers valuable information about the status of epithelial barrier integrity, the effectiveness of barrier-targeting therapeutics, and the pathophysiology of diseases.

4.1 Limitation for the evaluation of intestinal permeability

Despite the advantages that intestinal permeability tests, correlation results of these biomarkers and functional permeability through studies are difficult to interpret, representing a major shortcoming in the field. Due to a number of limitation mentioned in Tables 1 and 2 extrapolation of intestinal permeability findings to overall barrier function should be taken into consideration. Some major limitations include:

1. Limitation to detect acute intestinal damage.

2. Specificity and sensitivity of many biomarkers to detect small changes in intestinal permeability.

3. The outcome in most of these techniques is a single value of permeability, which does not allow to discern which region of the GI is being affected, and therefore must be used in combination with other methods [69].

4. Factors that affect the distribution and excretion of the orally administered probe, including gastric emptying, intestinal transit, bacterial degradation, intestinal blood flow [70].

5. Timing of blood or urinary collection should be taken into consideration [71].

Therefore, reliable, reproducible, and feasible methods for measuring and interpreted appropriately of intestinal permeability in the clinical setting are necessary.

Table 1. Measurements of intestinal permeability based on the assay.

Assay	Measurement	Probe	Limitations
In vivo	Oligosaccharides of different molecular weights [72]	Sugars	Baseline food contamination, laborious detection with HPLC, GC-MS or LC-MS
		Sucrose	Degraded by sucrose in the duodenum
		Lactulose	Degraded by intestinal bacteria
		Mannitol	Degraded by intestinal bacteria
	Multi sugar test with rhamnose [73]	Sucralose	Resistive to bacterial degradation Long collection time
		Macromolecules	Correlation with food-related antigens Limited use in humans
		Ovalbumin, HRP, dextrans	Invasive when used in intestinal loops
Ex vivo	Intestinal loop	Ions, sugars, PEG, FITC-dextran, OVA, 51Cr-EDTA, bacteria	Invasive
	Using chambers [74]	Ions, sugars, PEG, FITC-dextran, HRP, OVA, 51Cr-EDTA	Need for fresh tissue, limited viability (2 hours), laborious, operator-dependent
In vitro	Cell culture monolayers	Electrical resistance, drugs, FITC-dextran, PEG, 51Cr-EDTA, bacteria	Long follow-up times, wide range of molecules, different test conditions Less representative to the in vivo condition, cell culture variability Lacks host complexity

Table 2. Measurements of intestinal permeability based on biomarkers.

Biomarkers	Biological material	Limitations
Urine	Claudin proteins [72]	Non-specific for gut (e.g. release from kidney epithelia)
	FABP [75]	Only useful for acute damage
	α-GST [76]	Non-specific, possibly only useful for acute damage
Blood	FABP [75]	Only useful for acute damage
	α-GST [76]	Non-specific, possibly only useful for acute damage
	Citrulline [77]	Laborious detection
	Zonulin [78]	Low specificity for detection with ELISA
	Bacteria and bacterial products [69]	Not all bacteria can be cultured, high false positive rate for LPS detection
	EndoCab [79]	Indirect marker for bacterial translocation, correlation with IP conflicting
	D-lactate [80]	Limited data in humans Non-specific, influenced by bacterial overgrowth
Feces	Calprotectin [81]	Indirect marker dependent on the presence of intestinal inflammation

4.2 Future direction for the evaluation of the intestinal permeability

In recent years, a few excellent techniques have been introduced to evaluate intestinal barrier function, including confocal laser endomicroscopy (CLE) and primary epithelial monolayer cultures.

CLE, a novel endoscopic-assisted technique, has emerged as a valuable tool for gastrointestinal endoscopic imaging. New imaging techniques with confocal laser endomicroscopy provide exciting opportunities for visualization and quantification in order to evaluate intestinal barrier integrity *in vivo* [82] with visualizing epithelial cell shedding and sites of barrier loss in patients during endoscopy in real time [83]. Several conducted studies have shown that CLE identifies loss of the mucosal barrier in patients with colorectal cancer [84], gastric cancer [85] and inflammatory bowel disease [86]. Laser-scanning confocal fluorescence microscopy has become an indispensable tool for a wide range of biomedical investigations research by virtue of its high spatial resolution. Confocal microscopy is widely used to visualize, compare and evaluate the staining intensity of the tight junctional complexes in order to determine their expression.

Examination of images is a time-consuming and tedious work as a single scan for a patient may include up to tens of thousands of images of the GI tract. Moreover, abnormal frames may occupy only a very small portion of all of the images that might be missed due to fatigue or oversight [87]. In recent years, there has been a great interest in the computer image analysis in cell microscopy to develop automated algorithms to classify cell images into different grades (such as healthy, moderate and severe) [88], sub-cellular structures and studies on pattern recognition living cells [89]. The most important part of these techniques is based on extracting contextual features from given input images (e.g. edges, contours, morphology, and texture). The extracted features are then used in training deep learning models that are able to distinguish between different grades of the disease. Therefore deep-learning algorithms, such as CNN technique can be an important option for real-time evaluation of intestinal barrier function that showed to exceed human performance in visual tasks in a broad range of diseases [47]. The main contributions can be summarized in terms of the following four aspects:

(1) This work is the first, at the best of our knowledge, experimental study to include porcine intestinal dataset collections to explore the feasibility of deep convolutional neural network for barrier integrity diagnosis.

(2) Due to small size of dataset, the techniques for data augmentation including horizontal and vertical flips, rotating, zooming, contrast adjustment and brightness enhancement were applied to avoid the negative effect of the small data samples and to sufficiently train a CNN model. To reduce the heterogeneity of provided data, and hence, improvement of the classification performance, the pre-processing steps were used. These preprocessing steps also help to reduce the sensitivity of the deep CNN to contrast and intensity bias. Therefore, the deep CNN gained a high-level structure information more efficiently that ensures to gain a better prediction.

(3) An extensive comparison of different state-of-the-art CNNs provided to select the most promising networks for accurate barrier integrity recognition in IPEC-J2 images.

(4) A hybrid CNN model based on their great success in image classification for classifying IPEC-J2 is proposed.

(5) The goal of aggregating features from intermediate convolutional layers is to generate high-level discriminative feature maps from deep CNN architectures. This approach results in the detection of more complex patterns from each input image that gives a high accuracy with a low error rate.

5. The beneficial effect of deep convolutional neural network

The use of image processing technology has recently been extended to animal behavior and welfare research as it offers tools to monitor and to obtain feedback on animal location [90]. Although the effect of deep convolutional neural network (deep CNN) in other areas has long been investigated, its application in the animal experimental study, particularly in porcine intestinal cells, has not been yet investigated. It is expected to improve livestock animal health and welfare assessments to enhance problem identification in the livestock industry. However,

animal intestinal abnormality detection is difficult to analyze in practice, due to the inefficiencies involved in manually determining, and shortfall of experimental tests. Furthermore, the effectiveness of an abnormality assessment relies on the intuition of the observer, which may vary considerably between assessors. Hence, this study investigates the application of machine learning systems to recognize and detect the intestinal abnormality of animals in a quantitative manner. Findings indicate that further research is required to develop systems that can accurately detect the morphology and abnormality of animals more objectively and effectively in a commercially realistic environment.

Π. Introduction

Intestinal epithelial cells play an essential role in mucosal immunity that greatly contributes to the maintenance of the gastrointestinal system. Its role includes participating in the exchange of nutrients, secreting different regulatory mediators [1], constructing mucosal barriers, and responding to bacterial antigens [2]. This intestinal barrier constituted by tight junction proteins forms an integrated network of tissues, cells, and molecules that protect intestine against a wide variety of pathogens. The integrity of the gastrointestinal epithelial barrier is compromised in tight junction dysfunction. Recent studies have reported that the loss of gut integrity caused by disruption of thigh junction proteins could lead to the pathogenesis of many intestinal disorders [5]. In addition, activation of the mucosal immune system caused by loss of barrier integrity led to secretion of inflammatory mediators, which in turn result in increased barrier dysfunction.

Direct visualization of barrier disruption using confocal microscopy can provide a more detailed analysis of intestinal epithelial permeability. The use of lasers, a wide range of fluorescent probes and detectors also can aid in the analysis of multiple parameters within one sample. Several methods have been developed for non-invasive measurement of gut permeability by providing integrated measurement and have been successful in the examination of gut permeability of barrier function [91-

92]. While these techniques have proved useful, they do not provide optical information on the sites of barrier dysfunction.

Taking samples from abnormal or suspicious tissue is a pre-requisite step in studying the manifestations of the disease. With the recent technological advances, the speed for producing cell lines [93] has increased significantly and provided an unlimited supply of material associated with the use of animal and human that made a large-scale of histopathology images available that need to be analyzed. Besides, modern technology allows studying digital slides on a computer with a high resolution and magnification level rather than through a microscope. However, the manual examination and interpretation of acquired images is a challenging task since it is expensive, error-prone, time-consuming as well as the requirement of experienced pathologists. Furthermore, the inter- and intra-observer variability can adversely affect the reproducibility of the results [94].

Given the importance of monitoring intestinal permeability and the significant healthcare cost associated with the gut barrier disruption, early detection and interpretation in measuring intestinal permeability are an area of active research, and adequate experimental models are therefore required to further understand the gut barrier disruption [95]. Over the past decade, developing automated computer-aided diagnosis (CAD) systems that integrated with image processing and machine learning methods, has drawn considerable attention by achieving success on various biomedical image analysis tasks [94], and even outperforming the duty of the physicians [96], . These methods could be considered as the second opinion to decrease misdiagnosis (false positive and/or false negative) of errors rate and also reduce the heavy workload of manual diagnosis.

The main goal of the present study was to develop a deep learning method to automatically analyze digitized IPEC-J2 images. In this thesis, designing and evaluating different deep learning architectures for automatic feature extraction from input images in order to accurately assess the grade of localization and disruption of IPEC-J2 tight junction proteins has been focused. Therefore, developing an automated method based on deep learning models has the potential to provide

pathologists with a second opinion to decrease the misdiagnosis rate of IPEC-J2 images and also reduce the heavy workload of manual diagnosis.

III. Materials and methods

1. Methodology

The methodology used in this study is mainly divided into the following six stages, as demonstrated in Figure 4. Initially, the IPEC-J2 cell images were collected to make dataset. Then, different data augmentation methods such as vertical flip, horizontal flip, rotation, illumination and contrast enhancement have been employed to increase the dataset size. After that, different necessary pre-processing methods specific for deep learning architecture are utilized to improve the quality of visual information of input image. All of these data preparation methods help to increase the visibility of crucial structures from each input image. After data preparation, input images fed into the selected deep CNN architectures to obtain initial results. Based on the achieved results, an ensemble CNN architecture designed to investigate the method of hybrid CNN architecture to furthermore the detection rate of the IPEC-J2 cells image classification. Finally, the performance of the proposed architecture is validated on test images.

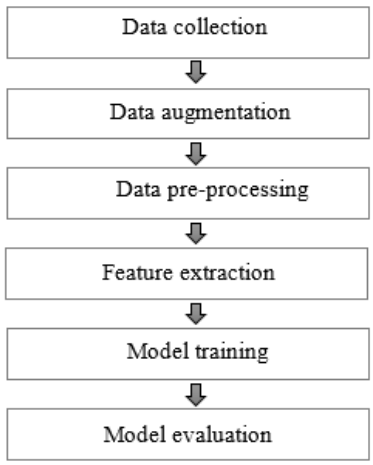


Figure 4. The block diagram of the proposed methodology.

2. Motivation and Contribution

2.1 The contribution of the proposed ensemble model:

The first, in this thesis, extracting a set of discriminative features from relatively small training data can achieve promising results is demonstrated. Additionally, the issue of small dataset with employing different data augmentation techniques is addressed. Different regularization strategies are applied such as L1 and L2 regularization to address overfitting issue and improve the generalizability of the proposed CNN architecture. The second, the performance of different well-established architectures in extracting high-level contextual information in order to boost the recognition of slight differences between classes in IPEC-J2 image classification is evaluated. The performance of 8 widely-used deep CNN architecture namely, VGG-Net [29], InceptionV3 [22], MobileNet [24], DenseNet [28], Xception [23], NAS-Net [26], InceptionResNetV2 [30] and ResNet [27] models are considered. Results indicate the ensemble of different CNN architectures and also the aggregation [100] of intermediate CNN layers leads to a better performance despite the small number of training samples. Moreover, a multi-layer perceptron classifier with three layers is used to train the features extracted from ensemble

models. Furthermore, a comparative study of the performance of different ensemble models with the individual deep CNN architectures demonstrates the benefits of the proposed approach.

Finally, the transfer learning strategy and an extensive hyper-parameter tuning are employed to find the optimal parameters in the training process, which are proved to be very beneficial for different image analysis tasks. Transferring the weights trained on ImageNet dataset as weights initialization instead of training our models using weights initialized randomly is another purpose for this study. The presented ensemble classification architecture obtains better results compared with that of individual deep CNN models. The metrics for performance evaluation of all ensemble and individual models are based on accuracy, sensitivity and specificity.

2.2 Two-path ensemble architecture for IPEC-J2 cell image classification

The main contribution of this research is to evaluate the performance of widely-used deep CNN architectures and then propose an ensemble CNN architecture that combines features from selected intermediate layers to generate discriminative feature for IPEC-J2 cells image classification. For designing the ensemble of deep CNN models, best architectures from the first stage is selected based on their i) ability in extracting discriminative features from the input images, ii) obtaining optimal detection rate, and iii) feasibility of employing transfer learning technique. Ensemble of deep learning models recently has drawn considerable attention due to its recent success in image analysis in comparison with individual deep CNNs. Since, each CNN architecture has its limitations in addressing the large variations of shape, texture and also the structure as present in IPEC-J2 cell images. Ensemble models also can improve the generalization ability of the network and improve the performance by combining the output result of each architecture.

With aforementioned advantages of the ensemble models, in this study, different two-path ensemble models are proposed to make use of the benefits of the best classifiers. I proposed an ensemble of InceptionV3 and DenSeNet201 architectures as an example for IPEC-J2 cells image classification. Referring to Figure 5, the

proposed model is constructed based on two pre-trained CNN architectures as a feature extractor. Features are immediately extracted from convolutional layers after each pooling layers as auxiliary supervision features for further improvement. Next, a global average pooling is applied on each selected intermediate layers. These auxiliary features from intermediate layers can aid in learning high-level discriminative features besides the existing extracted features from individual pre-trained architectures. The extracted features from each architecture and also auxiliary supervision features then are concatenated together to produce the final feature vector. The final feature vector then used as input for training a multi-layer perceptron classifier with three fully connected layers consist of different hidden neurons to train extracted features. Finally, three output neurons associating with healthy, moderate and severe cases with softmax non-linearity activation function are used at the classifier layer.

Applying a global average pooling operation can aid in reducing the number of parameters and prevent getting stuck in the poor local minima in very deep CNNs architectures while the training dataset is small. Furthermore, by decreasing the number of parameters, the gradient can flow within the very deep network regardless of the network depth, i.e. the number of convolutional layers, hidden layers and units.

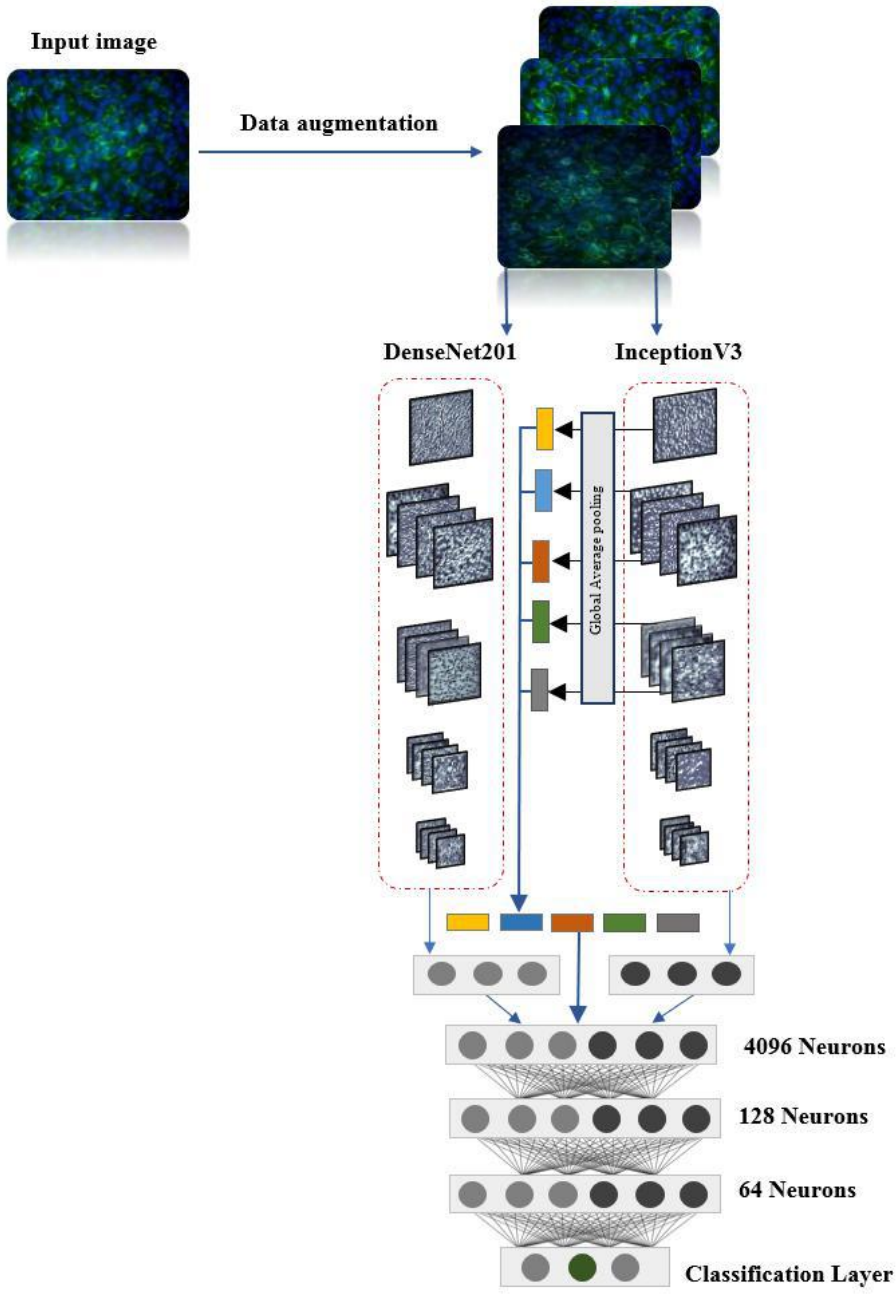


Figure 5. Illustration of the proposed ensemble network with a two-path CNNs of DenSeNet201 and InceptionV3.

IV. Experiment

1. Dataset description

The dataset used for this research collected from different studies in the literature. Informations for figures reuse permission are further detailed in the Tables 3 and 4, and Appendix 2.

Table 3. Copyright permissions of figures used for dataset.

	Figures	Name of work	Copyright and Permission	Ref.
1	Fig. 4 Fig. 5	Altered cytokine expression and barrier properties after *in vitro* infection of porcine epithelial cells with enterotoxigenic *Escherichia coli* and probiotic *Enterococcus faecium*	Licensed with a Creative Commons Attribution-NonCommercial-NoDerivs 2.0 Generic (CC BY-NC-ND 2.0)	[101]
2	Fig. 1B Fig. 2F	*Bacillus subtilis* protects porcine intestinal barrier from deoxynivalenol via improved zonula occludens-1 expression	Licensed with a Creative Commons Attribution-NonCommercial-3.0 Unported (CC BY-NC 3.0)	[102]
3	Fig. 3C Fig. 6C	Cecropin A modulates tight junction-related protein expression and enhances the barrier function of porcine intestinal epithelial cells by suppressing the MEK/ERK pathway	Licensed with a Creative Commons Attribution 4.0 International (CC BY 4.0)	[103]
4	Fig. 7	Effects of *Lactobacillus johnsonii* and *Lactobacillus reuteri* on gut barrier function and heat shock proteins in intestinal porcine epithelial cells	Licensed with a Creative Commons Attribution 4.0 International (CC BY 4.0)	[104]
5	Fig. 4C Fig. 8E	Nontoxic concentrations of OTA aggravate DON-induced intestinal barrier dysfunction in IPEC-J2 cells via activation of NF-κB signaling pathway	Licensed with the terms and conditions provided by Elsevier and Copyright Clearance Center.	[105]
6	Fig. 3B	The food contaminant deoxynivalenol, decreases intestinal barrier permeability and reduces claudin expression	Licensed with the terms and conditions provided by Elsevier and Copyright Clearance Center.	[106]
7	Fig. 4C Fig. 6A	Toll-like receptor 5-mediated IL-17C expression in intestinal epithelial cells enhances epithelial host defense against F4+ ETEC infection	Licensed with a Creative Commons Attribution 4.0 International (CC BY 4.0)	[107]

Table 4. Copyright permissions of figures used for dataset.

	Figures	Name of work	Copyright and Permission	Ref.
8	Fig. 7	Thymol improves barrier function and attenuates inflammatory responses in porcine intestinal epithelial cells during lipopolysaccharide (LPS)-induced inflammation	Copyright © 2019, American Chemical Society	[108]
9	Fig. 4	Protective effects of *Lactobacillus plantarum* on epithelial barrier disruption caused by enterotoxigenic *Escherichia coli* in intestinal porcine epithelial cells	Licensed with the terms and conditions provided by Elsevier and Copyright Clearance Center.	[109]
10	Fig. 1C,D Fig. 2F,H Fig. 3D Fig. 4E Fig. 5G	OTA induces intestinal epithelial barrier dysfunction and tight junction disruption in IPEC-J2 cells through ROS/Ca2þ-mediated MLCK activation	Licensed with the terms and conditions provided by Elsevier and Copyright Clearance Center.	[110]
11	Fig. 7 Fig. 8	Vulnerability of polarised intestinal porcine epithelial cells to mycotoxin deoxynivalenol depends on the route of application	Licensed with a Creative Commons Attribution 4.0 International (CC BY 4.0)	[111]
12	Fig. 7 Fig. 8 Fig. 9	β-Conglycinin-induced intestinal porcine epithelial cell damage via the nuclear factor κB/mitogen-activated protein kinase signaling pathway	Copyright © 2019, American Chemical Society	[112]
13	Fig. 4	β-Conglycinin reduces the tight junction occludin and ZO-1 expression in IPEC-J2	Licensed with a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0)	[113]
14	Fig. 1D Fig. 2D Fig. 4C,D	Barrier protection via Toll-like receptor 2 signaling in porcine intestinal epithelial cells damaged by deoxynivalenol	Copyright © 2016, Springer Nature	[114]

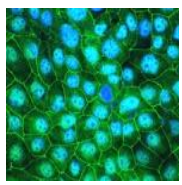
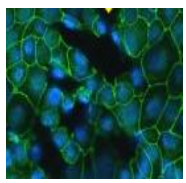
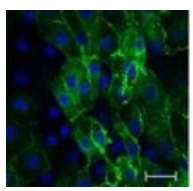
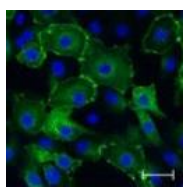
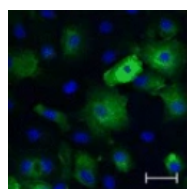
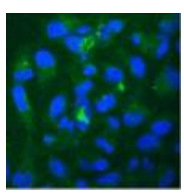
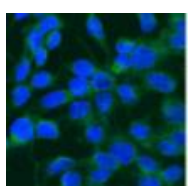
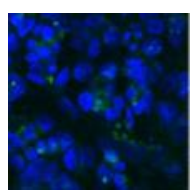
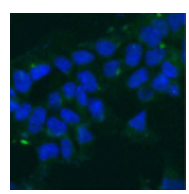
IPEC-J2 cells were treated with different stimuli including: enterotoxigenic *Escherichia coli* (ETEC) [3-6], Deoxynivalenol [7-10], Ochratoxin A (OTA) [115][110], Citrinin [115], Lipopolysaccharide (LPS) [116] and/or β-conglycinin [112]. Cellular distribution and Immunofluorescence localization of the tight junction proteins (Claudin, Occludin and ZO-1) were visualized and evaluated using confocal laser scanning microscopy.

The goal of this research is to develop a deep learning-based model to automatically classify cells within IPEC-J2 cell images. The dataset consists of 102 IPEC-J2 cell images acquired with a fluorescence microscope. IPEC-J2 cells are categorized into three classes: Normal or untreated (44 cell images), Moderate (28 cell images), and Severe (30 cell images). Representative figures of Normal, medium and severe TJs' disruption in the IPEC-J2 cells are depicted in Figures 6, 7 and 8, respectively. The first group of untreated IPEC-J2 cells was defined as the normal class, the two remain treated groups were categorized as a moderate (second) class and severe (third) class.

In the experiment using two-groups, first untreated IPEC-J2 cells were defined as the normal class and the second group were defined as a severe class.

In the experiment with four-group's treatment, first untreated IPEC-J2 cells were defined as normal, moderate, and two last groups were categorized as a severe class. The methodology of disturbing IPEC-J2 cells barrier integrity pattern is depicted in Table 5.

Table 5. IPEC-J2 cells barrier integrity annotation based on normal, moderate and severe TJs' disruption.

1					Ref.
	Two-group treatment				
	Control	Severe			
					[104]
2	**Three-group treatment**				
	Control	Moderate	Severe		
					[114]
3	**Four-group treatment**				
	Control	Moderate	Severe		
					[112]

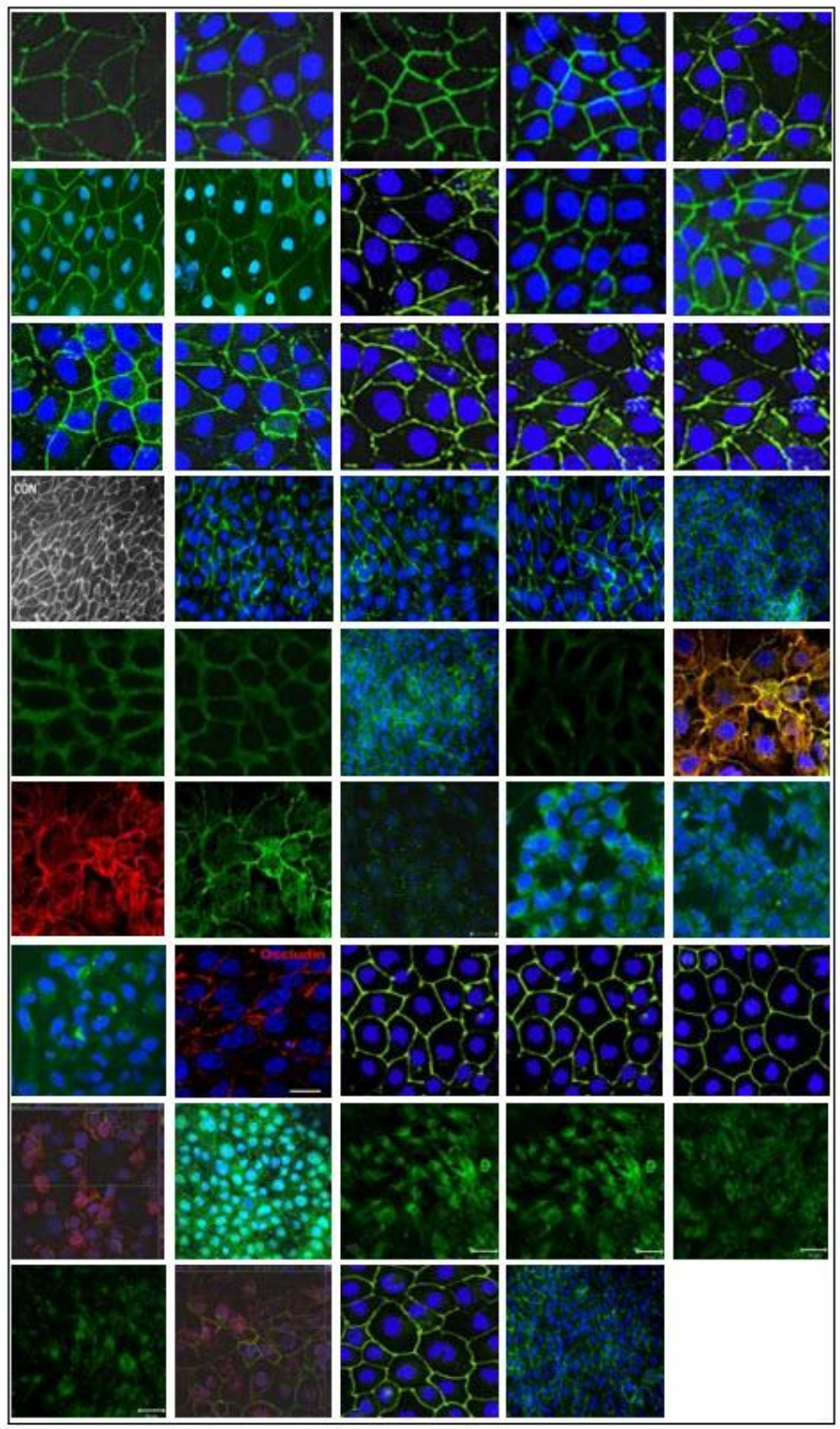


Figure 6. Representative figures showing normal TJs in IPEC-J2.

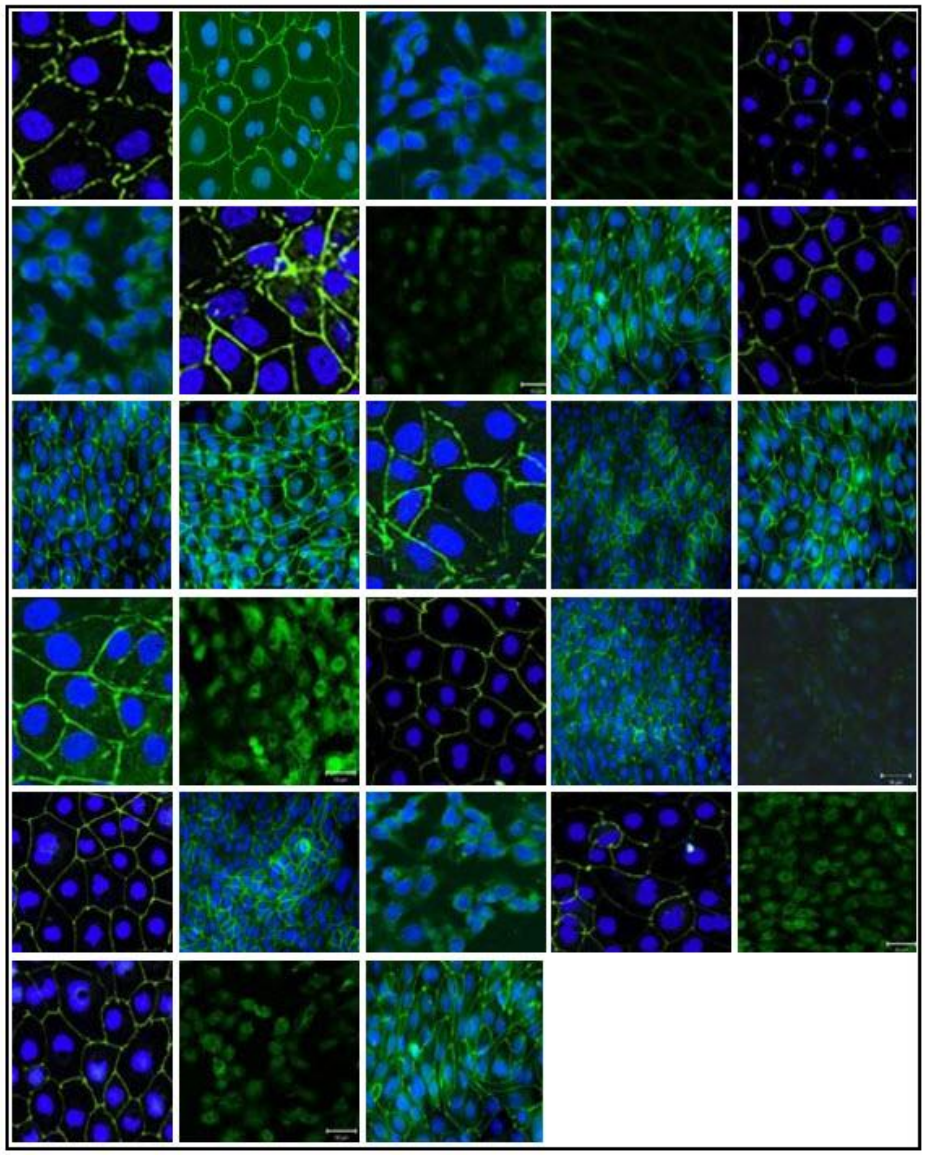


Figure 7. Representative figures showing medium disruption of TJ in IPEC-J2.

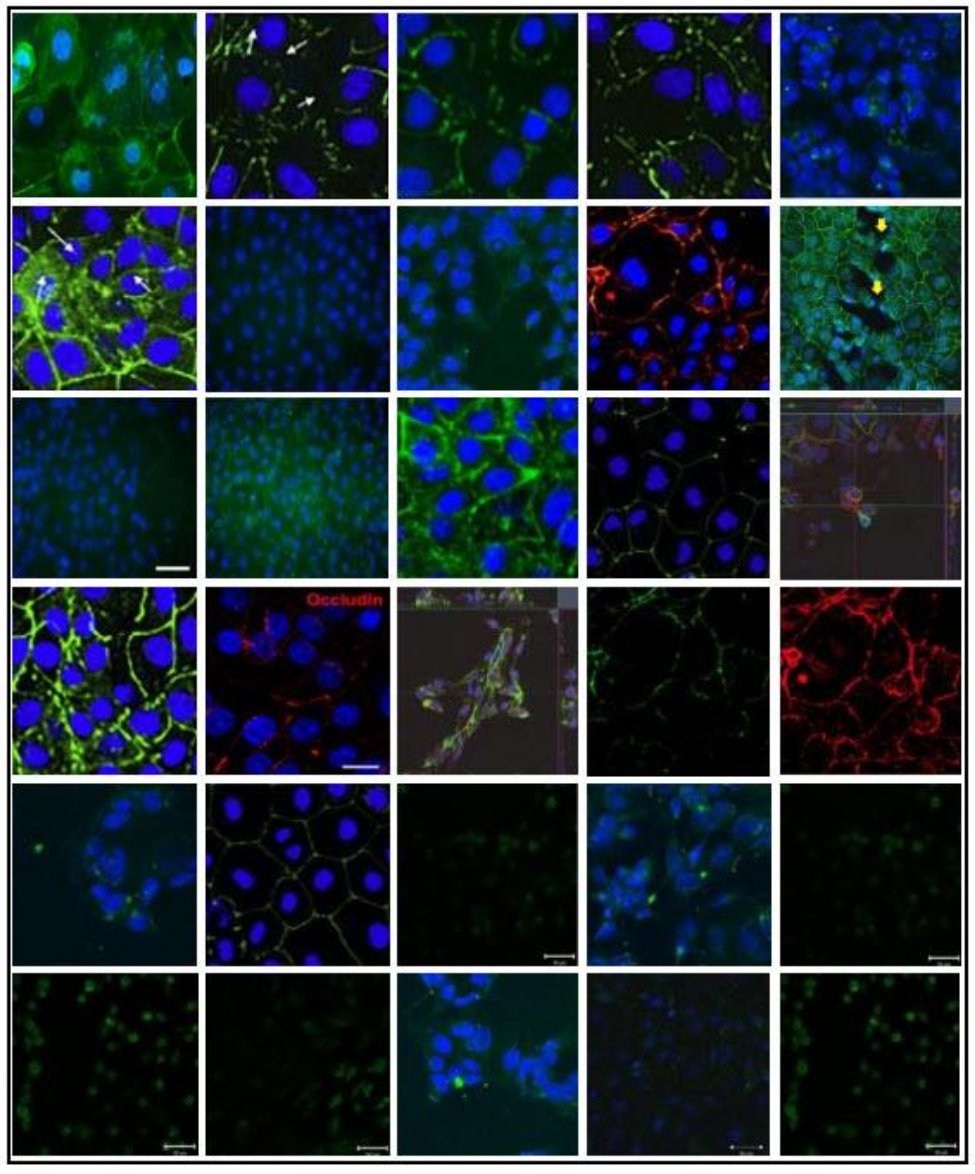


Figure 8. Representative figures showing severe disruption of TJ in IPEC-J2.

2. Data pre-processing

2.1 Resizing:

Concerning the variations in the dimension of the input images, all of the images have been resized from the center to the final resolution of 256×256 pixels according to the aspect ratio using bicubic interpolation.

2.2 Z-score image normalization:

In order to remove bias from the features, the intensity value of the all images have been normalized so as to have a zero mean and a standard deviation of one. Using Z-score image normalization, a uniform distribution of the dataset can be obtained.

2.3 Image normalization:

Another essential pre-processing step in deep learning models is image normalization [117]. In this way, all input images are normalized by min-max normalization to the intensity range of [0, 1], which is calculated as:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

where X is the pixel intensity. X_min and X_max are minimum and maximum intensity values of the input image, respectively. Images also have been normalized using ImageNet mean subtraction. The ImageNet mean is a pre-computed constant derived from ImageNet database [118].

3. Data augmentation

Due to the limited size of provided datasets, various data augmentation techniques applied to sufficiently train a CNN models. In this regard to address the issue of limited dataset size, different data augmentation methods, including horizontal and vertical flips, rotating, zooming, contrast adjustment and brightness enhancement are employed to avoid the negative effect of the small data samples and optimize the performance of the deep CNN model. The data augmentation parameters employed

for this research are presented in Table 6. Examples of images after applying different data augmentation techniques are shown in Figure 9.

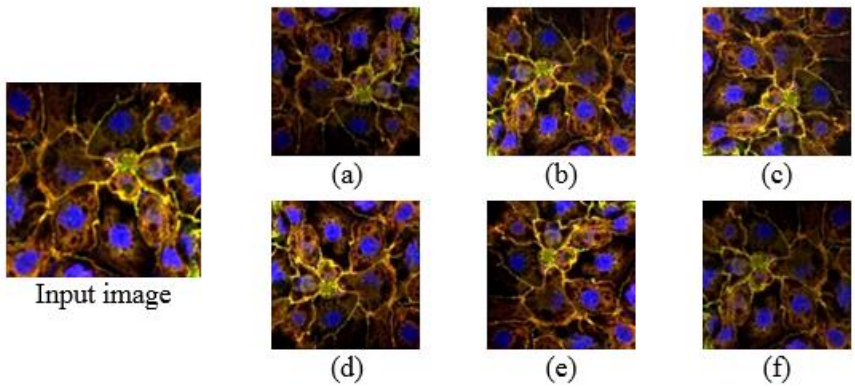


Figure 9. Images after data augmentation techniques. (a) rotation and contrast adjustments, (b) vertical, horizontal flip, (c) vertical, horizontal flip, contrast adjustments and rotation, (d) horizontal and vertical flip, contrast adjustments and brightness correction (e) vertical flip, brightness correction and rotation, (f) vertical flip, horizontal flip and contrast adjustments.

Table 6. Data augmentation parameters employed to increase the dataset size.

No.	Parameter	Value
1	Rotation	90
2	Zoom range	0.2
3	Shear range	0.3
4	Contrast adjustment	True
5	Brightness enhancement	True
6	Vertical flip	True
7	Horizontal flip	True
8	Fill mode	Nearest

4. Metrics for performance evaluation

To quantify the performance of the proposed method based on the obtained results, three evaluation metrics, namely, accuracy, sensitivity and specificity, are used[119]. Given the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN), the measures are mathematically expressed as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100$$

$$Sensitivity = \frac{TP}{TP + FN} \times 100$$

$$\text{Specificity} = \frac{TN}{TN + FP} \times 100$$

5. Experimental Setup

For implementing all of the experiments, 60% of the input images of each class are used for training set, 20% for the validation set, and the remaining 20% for the test set. All images were resized to the size of 256×256 pixels using bicubic interpolation. To achieve optimal performance, first perform an extensive grid-search for several hyper-parameter tuning performed to obtain the best hyper-parameter sets for training all models. The batch size was set to 64, and all models were trained for 400 epochs. The multi-layer perceptron classifier with three fully connected layers with 4096, 128 and 64 hidden neurons. The last fully connected layers followed by a dropout layer with the probability of 0.5 for a dropout layer to prevent over-fitting issue. Tanh activation function used for all fully connected layers and L1 and L2 regularization with values of 0.001 and 10-6, respectively. Adam optimizer is used to update the weights for all of the experiments with β1, β2 and learning rate of 0.6, 0.8 and 0.0001, respectively. The operating system is Windows with an Intel(R) Core (TM) i7-8700K 3.7 GHz processors with 32 GB RAM, and the proposed architecture implemented in Python using Keras package with Tensorflow as backend and run on Nvidia GeForce GTX 1080 Ti GPU with 11GB RAM.

V. Results

The obtained results are derived from the 765 test images, which were not used in the training phase. The test set was consisting 329 healthy cases, 211 moderate cases and 225 severe cases on the augmented dataset. The evaluation metrics of accuracy, sensitivity and specificity used to measure the performance of the models.

1. Deep features extraction based on transfer learning

The experiments started by investigating the effect of different deep CNN architectures in extracting deep features from input image and achieving optimal prediction on the unseen test data. As the results confirm, there is a level of variation in all results when running the experiments with different deep CNN architectures. Referring to Table 7, MobileNet, NASNetLarge and InceptionV3 with accuracies of 95.82%, 95.16% and 95.03%, respectively, significantly outperforms the other individual architectures on the provided dataset. The worst classifier is observed by ResNet50 architectures with an accuracy of 56.99%, sensitivity of 63.72%; however, the high specificity of 100%. High sensitivity (100%) result achieved by DenseNet201, and high specificity (100%) obtained by VGG19, ResNet50 and InceptionResNetV2 architectures.

Table 7. Classification results from pre-trained deep CNN models.

Method	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**
Xception	91.11	99.36	93.72
InceptionV3	95.03	97.50	93.54
MobileNet *	95.82	97.31	98.52
ResNet50	56.99	63.72	100
DenseNet121	86.41	78.36	97.34
DenseNet201	91.11	100	80.22
InceptionResNetV2	80.78	70.99	100
VGG16	81.05	73.13	98.83
VGG19	82.35	75.40	100
NASNetLarge $	95.16	99.35	91.15
NASNetMobile	90.46	95.25	87.50

* The best result; $ represents the second-best result of the respective category.

2. Deep feature extraction based on deep learning-based ensemble models

Various individual deep CNN architectures with different types of convolutional modules and different number of convolutional layers and parameters were selected for this thesis. However, few studies presented on the application of ensemble models in the literature [120-122]. Hence, the potential of deep CNN ensemble of the best performing models from results from pre-trained deep CNN models will be explored to further improve the final performance. Based on the results, architectures have been selected based on their good prediction performance. Then, different two-path ensemble-based methods with appropriate fine-tuned parameters were designed for more experiments. Based on the results derived from Table 7, the performance of 5 architectures, namely, MobileNet, NASNetLarge, InceptionV3, DenseNet201 and Xception architectures, achieve the best results; hence, other architectures have been dismissed for subsequent experiments. Nine version of two-path ensemble models, as showed in Table 8, were designed using these selected 5 architectures. The obtained results from different deep CNN ensemble architectures are presented in Table 8. Referring to Table 8, it is clear that some deep learning-based ensemble methods obtained better predictions than the individual models. On comparing the performance of the individual and ensemble architectures, it is found that the ensemble of InceptionV3 and DenseNet201 achieved the best result with an accurate detection rate of 99.22% than the individual InceptionV3 architecture with detection rate of 95.03% and the individual DenseNet201 architecture with detection rate of 91.11%. The second-best ensemble architecture, the ensemble of InceptionV3 and MobileNet, also yields a higher overall accuracy with detection rate of 97.78% than the individual InceptionV3 architecture with detection rate of 95.03% and the individual MobileNet architecture with detection rate of 95.82%. As an example, comparing deep CNN ensemble model of InceptionV3 with DenseNet201with the ensemble of InceptionV3 with MobileNet, it could gain 1.44% gap in terms of accuracy, 3.46% gap in terms of sensitivity, and both ensemble models achieved 100% for specificity. Similar conclusions can be drawn for other ensemble models.

Table 8. Classification results from different ensemble deep CNN models.

Method	**Accuracy (%)**	**Sensitivity (%)**	**Specificity (%)**
InceptionV3+DenseNet201 *	99.22	99.09	100
DenseNet201+NASNetLarge	43.01	60.92	-
InceptionV3+MobileNet $	97.78	95.63	100
InceptionV3+NASNetLarge	78.82	69.11	100
InceptionV3+Xception	82.88	100	92.13
MobileNet+DenseNet201	50.85	61.30	-
MobileNet+NASNetLarge	50.46	64.60	-
MobileNet+Xception	57.39	61.81	-
Xception+NASNetLarge	43.01	60.92	-

* The best result; $ represents the second-best result of the respective category.

The quantitative results for these two ensemble models are also summarized in the form of confusion matrices in Figure 10 and Figure 11. It can be seen that the misclassified cells of the ensemble of InceptionV3 and DenseNet201architecture are more commonly healthy cases that mislabeled as Moderate and Severe cells. In conclusion, the InceptionV3 with DenseNet201 is the best learner, and its counterpart, InceptionV3 with MobileNet, is the second-best learner. Referring to Table 8, it is clear that the ensemble of InceptionV3 with DenseNet201 architectures surpass other ensemble models in terms of accuracy (99.22%), sensitivity (99.09%) and specificity (100%), which implies that ensemble-based learning could extract more discriminative contextual features from the input images, hence leading to a better detection rate. In IPEC-J2 cell image classification on the test set, the ensemble of Xception with NASNetLarge obtained a very poor result with an accuracy of 43.01% and a sensitivity of 60.92%.

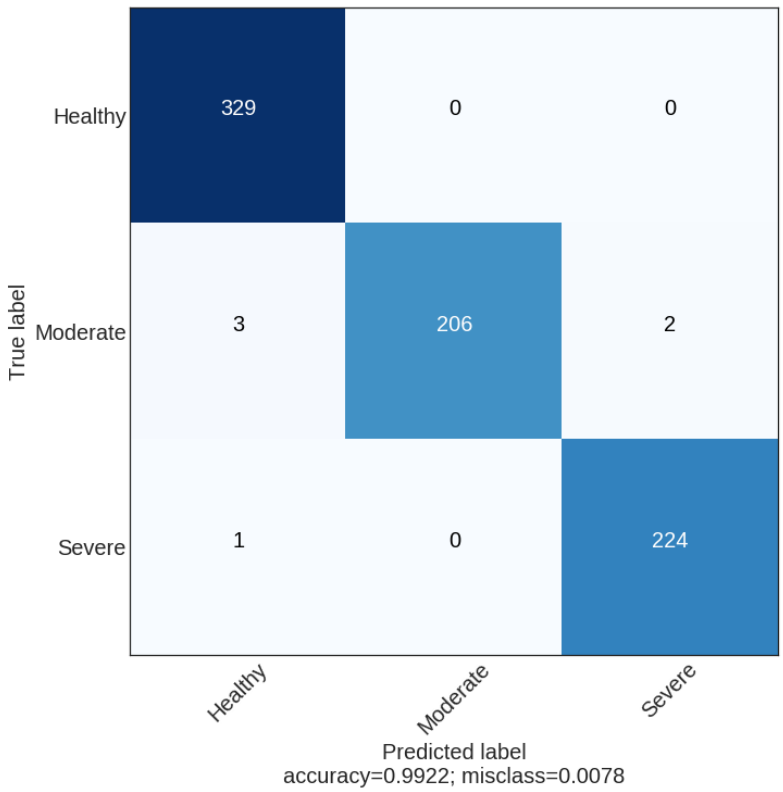


Figure 10. Confusion matrix of IPEC-J2 cell image classification using an ensemble of InceptionV3 and DenseNet201 model.

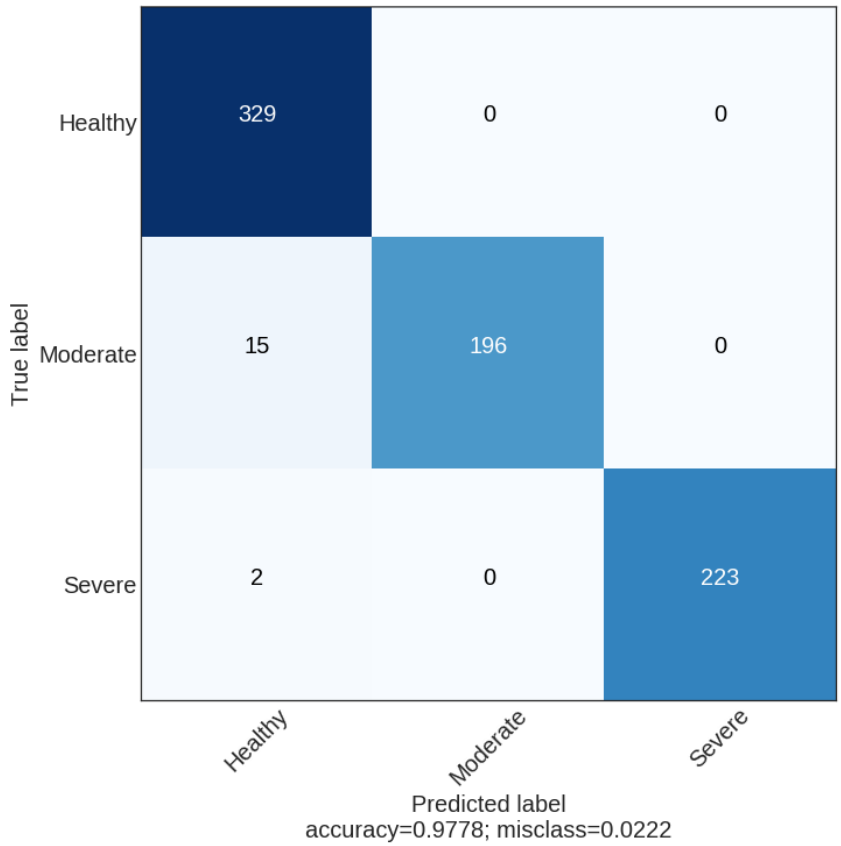


Figure 11. Confusion matrix of IPEC-J2 cell image classification using an ensemble of InceptionV3 and MobileNet model.

VI. Discussion

In this thesis, an effective two-path way deep learning-based ensemble method has been proposed for classification of IPEC-J2 cell image. The performance of various state-of-the-art deep learning architectures individually, as well as ensemble architectures, were evaluated and the ensemble of two well-established CNN architectures, namely, InceptionV3 and DenseNet201, achieved the highest accuracies and outperformed both the counterpart ensemble models and also individual CNN models. The extracted features from multiple abstraction levels of convolutional layers extracted as auxiliary supervision features for further improvement of the detection rates. Fusing features from selected intermediate layers can aid in learning high-level features besides the existing extracted features. Each of the selected deep CNN architectures for the present study is constructed by different types of modules and convolution, such as dense modules, inception module, residual blocks, depth-wise and point-wise layers with different learning ability can help learning different level of features from a given input image.

Based on the obtained results presented in Table 8, it can conclude that the proposed ensemble method achieved better results in terms of evaluation metrics of accuracy, sensitivity, and specificity than individual pre-trained state-of-the-art architectures for IPEC-J2 cell image classification. In this thesis, various well-established Deep CNN models employed. All of the models are trained with 4611 training images and validated on 1152 images on the augmented dataset. The performance of the proposed deep CNN ensemble models is evaluated on the 765 test images, which were not use in the training and validation phase. Different results are obtained from the selected CNN architectures based on the type of modules and convolutional structure and layers. Based on the obtained results from the first stage of this thesis, classifiers with lower accuracy rate have been excluded. Next, based on the results derived from Table 7, different ensemble models designed and implemented from the best individual architectures to investigate the advantages of utilizing the ensemble models with auxiliary supervision in improving the detection rate. The best ensemble CNN architecture achieved an overall accuracy to 99.22%, which is significantly higher than individual CNN architectures. This behavior was expected

due to the inception and dense models in both InceptionV3 and DenseNet201 architectures, which aids in extracting fine-grained contextual features to differentiate between different classes directly from input images more accurately. On comparing the performance of the ensemble architectures, it is found that the ensemble of InceptionV3 and DenseNet201 and also the ensemble of InceptionV3 and MobileNet has superior detection rates than other ensemble models and also individual architecture.

Experimental results demonstrate the two-path deep CNN ensemble architecture using pre-trained InceptionV3, and DenseNet201 architectures with the help of features aggregated from intermediate convolutional layers outperform all other deep CNN architectures and counterpart ensemble models with 99.20% in terms of accuracy.

VII. Literature cited

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, 1998.

[2] W. P. Amorim, E. C. Tetila, H. Pistori, and J. P. Papa, "Semi-supervised learning with convolutional neural networks for UAV images automatic recognition," *Comput. Electron. Agric.*, vol. 164, p. 104932, Sep. 2019.

[3] A. B. Levine, C. Schlosser, J. Grewal, R. Coope, S. J. M. Jones, and S. Yip, "Rise of the Machines: Advances in Deep Learning for Cancer Diagnosis," *Trends in Cancer*. 2019.

[4] M. Buda, A. Saha, and M. A. Mazurowski, "Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm," *Comput. Biol. Med.*, vol. 109, pp. 218–225, Jun. 2019.

[5] K. Roy, D. Banik, D. Bhattacharjee, and M. Nasipuri, "Patch-based system for Classification of Breast Histology images using deep learning," *Comput. Med. Imaging Graph.*, 2019.

[6] H. H. N. Pham, M. Futakuchi, A. Bychkov, T. Furukawa, K. Kuroda, and J. Fukuoka, "Detection of lung cancer lymph node metastases from whole-slide histopathological images using a two-step deep learning approach," *Am. J. Pathol.*, Sep. 2019.

[7] A. Dascalu and E. O. David, "Skin cancer detection by deep learning and sound analysis algorithms: A prospective clinical study of an elementary dermoscope," *EBioMedicine*, vol. 43, pp. 107–113, May 2019.

[8] H. H. Aghdam and E. J. Heravi, "Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification," in *Guide to Convolutional Neural Networks*, 2017.

[9] Y. D. Zhang, C. Pan, X. Chen, and F. Wang, "Abnormal breast identification by nine-layer convolutional neural network with parametric

rectified linear unit and rank-based stochastic pooling," *J. Comput. Sci.*, 2018.

[10] A. Acevedo, S. Alférez, A. Merino, L. Puigví, and J. Rodellar, "Recognition of peripheral blood cell images using convolutional neural networks," *Comput. Methods Programs Biomed.*, vol. 180, p. 105020, 2019.

[11] G. Caspari and P. Crespo, "Convolutional neural networks for archaeological site detection – Finding 'princely' tombs," *J. Archaeol. Sci.*, 2019.

[12] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010.

[13] X. Zhu, Z. Cai, J. Wu, Y. Cheng, and Q. Huang, "Convolutional neural network based combustion mode classification for condition monitoring in the supersonic combustor," *Acta Astronaut.*, 2019.

[14] S. Lu, Z. Lu, and Y. D. Zhang, "Pathological brain detection based on AlexNet and transfer learning," *J. Comput. Sci.*, 2019.

[15] L. H. S. Vogado, R. M. S. Veras, F. H. D. Araujo, R. R. V. Silva, and K. R. T. Aires, "Leukemia diagnosis in blood slides using transfer learning in CNNs and SVM for classification," *Eng. Appl. Artif. Intell.*, 2018.

[16] Z. N. K. Swati *et al.*, "Brain tumor classification for MR images using transfer learning and fine-tuning," *Comput. Med. Imaging Graph.*, 2019.

[17] A. S. Lundervold and A. Lundervold, "An overview of deep learning in medical imaging focusing on MRI," *Zeitschrift fur Medizinische Physik*, 2019.

[18] M. Santin *et al.*, "Detecting abnormal thyroid cartilages on CT using deep learning," *Diagn. Interv. Imaging*, 2019.

[19] P. Chen, L. Gao, X. Shi, K. Allen, and L. Yang, "Fully automatic knee

osteoarthritis severity grading using deep neural networks with a novel ordinal loss," *Comput. Med. Imaging Graph.*, 2019.

[20] X. Liu, C. Wang, J. Bai, and G. Liao, "Fine-tuning Pre-trained Convolutional Neural Networks for Gastric Precancerous Disease Classification on Magnification Narrow-band Imaging Images," *Neurocomputing*, 2019.

[21] F. van der Sommen, S. Zinger, E. J. Schoon, and P. H. N. de With, "Supportive automatic annotation of early esophageal cancer using local gabor and color features," *Neurocomputing*, 2014.

[22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," Dec. 2015.

[23] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017.

[24] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017.

[25] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017.

[26] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning Transferable Architectures for Scalable Image Recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017.

[29] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *CoRR*, vol. abs/1409.1, 2014.

[30] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-ResNet and the impact of residual connections on learning,” in *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, 2017.

[31] A. M. Marchiando, W. V. Graham, and J. R. Turner, “Epithelial Barriers in Homeostasis and Disease,” *Annu. Rev. Pathol. Mech. Dis.*, 2010.

[32] C. T. Kåhrström, N. Pariente, and U. Weiss, “Intestinal microbiota in health and disease,” *Nature*, vol. 535, no. 7610, pp. 47–47, Jul. 2016.

[33] T. Pelaseyed *et al.*, “The mucus and mucins of the goblet cells and enterocytes provide the first defense line of the gastrointestinal tract and interact with the immune system,” *Immunological Reviews*. 2014.

[34] M. Vancamelbeke and S. Vermeire, “The intestinal barrier: a fundamental role in health and disease,” *Expert Review of Gastroenterology and Hepatology*. 2017.

[35] L. R. Muniz, C. Knosp, and G. Yeretssian, “Intestinal antimicrobial peptides during homeostasis, infection, and disease,” *Frontiers in Immunology*. 2012.

[36] J. D. Schulzke, C. J. Bentzel, I. Schulzke, E. O. Riecken, and M. Fromm, “Epithelial tight junction structure in the jejunum of children with acute and treated celiac sprue,” *Pediatr. Res.*, 1998.

[37] J. Chang, R. W. Leong, V. C. Wasinger, M. Ip, M. Yang, and T. G. Phan, “Impaired Intestinal Permeability Contributes to Ongoing Bowel Symptoms in Patients With Inflammatory Bowel Disease and Mucosal Healing,” *Gastroenterology*, 2017.

[38] S. Ludidi *et al.*, “The intestinal barrier in irritable bowel syndrome: Subtype-specific effects of the systemic compartment in an In Vitro model,” *PLoS One*, 2015.

[39] L. Genser *et al.*, “Increased jejunal permeability in human obesity is revealed by a lipid challenge and is linked to inflammation and type 2 diabetes,” *J. Pathol.*, 2018.

[40] C. Perrier and B. Corthésy, “Gut permeability and food allergies,” *Clin. Exp. Allergy*, 2011.

[41] S. Einarsson, Y. Brandt, N. Lundeheim, and A. Madej, “Stress and its influence on reproduction in pigs: A review,” *Acta Veterinaria Scandinavica*. 2008.

[42] Y. Li, Z. Song, K. A. Kerr, and A. J. Moeser, “Chronic social stress in pigs impairs intestinal barrier and nutrient transporter function, and alters neuro-immune mediator and receptor expression,” *PLoS One*, vol. 12, no. 2, 2017.

[43] C. H. Hu, K. Xiao, Z. S. Luan, and J. Song, “Early weaning increases intestinal permeability, alters expression of cytokine and tight junction proteins, and activates mitogen-activated protein kinases in pigs,” *J. Anim. Sci.*, 2013.

[44] A. L. Ziegler and A. T. Blikslager, “Impaired intestinal barrier function and relapsing digestive disease: Lessons from a porcine model of early life stress,” *Neurogastroenterology and Motility*. 2017.

[45] N. K. Gabler, D. Koltes, S. Schaumberger, G. R. Murugesan, and N. Reisinger, “Diurnal heat stress reduces pig intestinal integrity and increases endotoxin translocation,” *Transl. Anim. Sci.*, 2018.

[46] Y. Belkaid and T. W. Hand, “Role of the microbiota in immunity and inflammation,” *Cell*. 2014.

[47] A. Perez-Lopez, J. Behnsen, S. P. Nuccio, and M. Raffatellu, “Mucosal immunity to pathogenic intestinal bacteria,” *Nature Reviews Immunology*. 2016.

[48] M. M. France and J. R. Turner, “The mucosal barrier at a glance,” *J. Cell Sci.*, 2017.

[49] J. M. Anderson and C. M. Van Itallie, “Tight junctions and the molecular basis for regulation of paracellular permeability,” *American Journal of Physiology - Gastrointestinal and Liver Physiology*. 1995.

[50] J. R. Turner, “Intestinal mucosal barrier function in health and disease,” *Nature Reviews Immunology*. 2009.

[51] C. M. Van Itallie and J. M. Anderson, “Claudin interactions in and out of the tight junction,” *Tissue Barriers*, 2013.

[52] A. Buckley and J. R. Turner, “Cell biology of tight junction barrier regulation and mucosal disease,” *Cold Spring Harb. Perspect. Biol.*, 2018.

[53] M. Beatch, L. A. Jesaitis, W. J. Gallin, D. A. Goodenough, and B. R. Stevenson, “The tight junction protein ZO-2 contains three PDZ (PSD-95/discs- large/ZO-1) domains and an alternatively spliced region,” *J. Biol. Chem.*, 1996.

[54] M. Itoh, M. Furuse, K. Morita, K. Kubota, M. Saitou, and S. Tsukita, “Direct binding of three tight junction-associated MAGUKs, ZO-1, ZO-2, and ZO-3, with the COOH termini of claudins,” *J. Cell Biol.*, 1999.

[55] H. Bauer, J. Zweimueller-Mayer, P. Steinbacher, A. Lametschwandtner, and H. C. Bauer, “The dual role of zonula occludens (ZO) proteins,” *Journal of Biomedicine and Biotechnology*. 2010.

[56] Y. Yoshida, Y. Ban, and S. Kinoshita, “Tight junction transmembrane protein claudin subtype expression and distribution in human corneal and conjunctival epithelium,” *Investig. Ophthalmol. Vis. Sci.*, 2009.

[57] S. Aijaz, M. S. Balda, and K. Matter, “Tight junctions: Molecular architecture and function,” *International Review of Cytology*. 2006.

[58] M. A. Odenwald *et al.*, “ZO-1 interactions with F-actin and occludin direct epithelial polarization and single lumen specification in 3D culture,” *J. Cell Sci.*, 2017.

[59] R. J. Mrsny *et al.*, “A key claudin extracellular loop domain is critical for

epithelial barrier integrity," *Am. J. Pathol.*, 2008.

[60] R. Noth *et al.*, "Increased intestinal permeability and tight junction disruption by altered expression and localization of occludin in a murine graft versus host disease model," *BMC Gastroenterol.*, 2011.

[61] R. Al-Sadi, K. Khatib, S. Guo, D. Ye, M. Youssef, and T. Ma, "Occludin regulates macromolecule flux across the intestinal epithelial tight junction barrier," *Am. J. Physiol. - Gastrointest. Liver Physiol.*, 2011.

[62] L. L. Mitic, C. M. Van Itallie, and J. M. Anderson, "Molecular physiology and pathophysiology of tight junctions I. Tight junction structure and function: Lessons from mutant animals and proteins," *American Journal of Physiology - Gastrointestinal and Liver Physiology*. 2000.

[63] V. Garcia-Hernandez, M. Quiros, and A. Nusrat, "Intestinal epithelial claudins: Expression and regulation in homeostasis and inflammation," *Annals of the New York Academy of Sciences*. 2017.

[64] S. Dabrowski *et al.*, "Redox-sensitive structure and function of the first extracellular loop of the cell-cell contact protein claudin-1: Lessons from molecular structure to animals," *Antioxidants Redox Signal.*, 2015.

[65] J. Nomme *et al.*, "Structural basis of a key factor regulating the affinity between the zonula occludens first PDZ domain and claudins," *J. Biol. Chem.*, 2015.

[66] L. Ding *et al.*, "Inflammation and disruption of the mucosal architecture in claudin-7-deficient mice," *Gastroenterology*, 2012.

[67] G. Krause, L. Winkler, C. Piehl, I. Blasig, J. Piontek, and S. L. Müller, "Structure and function of extracellular claudin domains," *Ann. N. Y. Acad. Sci.*, 2009.

[68] S. C. Bischoff *et al.*, "Intestinal permeability - a new target for disease prevention and therapy," *BMC Gastroenterology*. 2014.

[69] H. J. Galipeau and E. F. Verdu, "The complex task of measuring intestinal

permeability in basic and clinical science," *Neurogastroenterology and Motility*. 2016.

[70] L. Wang, C. Llorente, P. Hartmann, A. M. Yang, P. Chen, and B. Schnabl, "Methods to determine intestinal permeability and bacterial translocation during liver disease," *Journal of Immunological Methods*. 2015.

[71] I. Bjarnason, A. Macpherson, and D. Hollander, "Intestinal permeability: An overview," *Gastroenterology*, vol. 108, no. 5, pp. 1566–1581, May 1995.

[72] J. Grootjans, "Non-invasive assessment of barrier integrity and function of the human gut," *World J. Gastrointest. Surg.*, 2010.

[73] J. Stappaerts, J. Brouwers, P. Annaert, and P. Augustijns, "In situ perfusion in rodents to explore intestinal drug absorption: Challenges and opportunities," *International Journal of Pharmaceutics*. 2015.

[74] H. H. Ussing and K. Zerahn, "Active transport of sodium as th?? source of electric current in the short-circuited isolated frog skin," *J. Am. Soc. Nephrol.*, 1999.

[75] M. Furuhashi and G. S. Hotamisligil, "Fatty acid-binding proteins: Role in metabolic diseases and potential as drug targets," *Nature Reviews Drug Discovery*. 2008.

[76] S. Khurana, M. T. Corbally, F. Manning, T. Armenise, B. Kierce, and C. Kilty, "Glutathione S-transferase: A potential new marker of intestinal ischemia," *J. Pediatr. Surg.*, 2002.

[77] P. Crenn, B. Messing, and L. Cynober, "Citrulline as a biomarker of intestinal failure due to enterocyte mass reduction," *Clinical Nutrition*. 2008.

[78] A. Sapone *et al.*, "Zonulin upregulation is associated with increased gut permeability in subjects with type 1 diabetes and their relatives," *Diabetes*, 2006.

[79] J. C. Penalva *et al.*, “A study of intestinal permeability in relation to the inflammatory response and plasma endocab IgM levels in patients with acute pancreatitis,” *J. Clin. Gastroenterol.*, 2004.

[80] X.-Q. Sun, “Relationship between plasma D(-)-lactate and intestinal damage after severe injuries in rats,” *World J. Gastroenterol.*, vol. 7, no. 4, p. 555, 2001.

[81] S. Ikhtaire, M. S. Shajib, W. Reinisch, and W. I. Khan, “Fecal calprotectin: its scope and utility in the management of inflammatory bowel disease,” *Journal of Gastroenterology*. 2016.

[82] R. Kiesslich *et al.*, “Identification of Epithelial Gaps in Human Small and Large Intestine by Confocal Endomicroscopy,” *Gastroenterology*, 2007.

[83] R. Kiesslich *et al.*, “Local barrier dysfunction identified by confocal laser endomicroscopy predicts relapse in inflammatory bowel disease,” *Gut*, 2012.

[84] X. J. Xie *et al.*, “Differentiation of colonic polyps by confocal laser endomicroscopy,” *Endoscopy*, 2011.

[85] S. Kitabatake *et al.*, “Confocal endomicroscopy for the diagnosis of gastric cancer in vivo,” *Endoscopy*, 2006.

[86] L. G. Lim *et al.*, “Confocal endomicroscopy identifies loss of local barrier function in the duodenum of patients with Crohn’s disease and ulcerative colitis,” *Inflamm. Bowel Dis.*, 2014.

[87] B. Li and M. Q. H. Meng, “Computer-aided detection of bleeding regions for capsule endoscopy images,” *IEEE Trans. Biomed. Eng.*, 2009.

[88] J. C. Caicedo *et al.*, “Data-analysis strategies for image-based cell profiling,” *Nat. Methods*, 2017.

[89] S. Uchida, “Image processing and recognition for biological images,” *Development Growth and Differentiation*. 2013.

[90] R. D. Tillett, C. M. Onyango, and J. A. Marchant, “Using model-based image processing to track animal movements,” *Comput. Electron. Agric.*, 1997.

[91] R. B. Dorshow *et al.*, “Measurement of gut permeability using fluorescent tracer agent technology,” *Sci. Rep.*, 2017.

[92] M. González-González, C. Díaz-Zepeda, J. Eyzaguirre-Velásquez, C. González-Arancibia, J. A. Bravo, and M. Julio-Pieper, “Investigating gut permeability in animal models of disease,” *Frontiers in Physiology*. 2019.

[93] G. Kaur and J. M. Dufour, “Cell lines: Valuable tools or useless artifacts.,” *Spermatogenesis*, 2012.

[94] D. Allende *et al.*, “Inter-observer and intra-observer variability in the diagnosis of dysplasia in patients with inflammatory bowel disease: Correlation of pathological and endoscopic findings,” *Color. Dis.*, 2014.

[95] M. Camilleri, K. Madsen, R. Spiller, B. G. Van Meerveld, and G. N. Verne, “Intestinal barrier function in health and gastrointestinal disease,” *Neurogastroenterology and Motility*. 2012.

[96] G. Hinton, “Deep learning-a technology with the potential to transform health care,” *JAMA - Journal of the American Medical Association*. 2018.

[97] L. Yu, H. Chen, Q. Dou, J. Qin, and P. A. Heng, “Automated Melanoma Recognition in Dermoscopy Images via Very Deep Residual Networks,” *IEEE Trans. Med. Imaging*, 2017.

[98] R. C. Maron *et al.*, “Systematic outperformance of 112 dermatologists in multiclass skin cancer image classification by convolutional neural networks,” *Eur. J. Cancer*, 2019.

[99] C. Affonso, A. L. D. Rossi, F. H. A. Vieira, and A. C. P. de L. F. de Carvalho, “Deep learning for biological image classification,” *Expert Syst. Appl.*, 2017.

[100] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, “Deep Layer Aggregation,”

in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018.

[101] M. Kern, D. Günzel, J. R. Aschenbach, K. Tedin, A. Bondzio, and U. Lodemann, "Altered Cytokine Expression and Barrier Properties after In Vitro Infection of Porcine Epithelial Cells with Enterotoxigenic Escherichia coli and Probiotic Enterococcus faecium," *Mediators Inflamm.*, 2017.

[102] M. J. Gu, S. K. Song, S. M. Park, I. K. Lee, and C. H. Yun, "Bacillus subtilis protects porcine intestinal barrier from deoxynivalenol via improved zonula occludens-1 expression," *Asian-Australasian J. Anim. Sci.*, 2014.

[103] Z. Zhai *et al.*, "Cecropin a modulates tight junction-related protein expression and enhances the barrier function of porcine intestinal epithelial cells by suppressing the MEK/ERK pathway," *Int. J. Mol. Sci.*, 2018.

[104] H. Y. Liu *et al.*, "Effects of Lactobacillus johnsonii and Lactobacillus reuteri on gut barrier function and heat shock proteins in intestinal porcine epithelial cells," *Physiol. Rep.*, 2015.

[105] C. Ying, W. Hong, Z. Nianhui, W. Chunlei, H. Kehe, and P. Cuiling, "Nontoxic concentrations of OTA aggravate DON-induced intestinal barrier dysfunction in IPEC-J2 cells via activation of NF-κB signaling pathway," *Toxicol. Lett.*, 2019.

[106] P. Pinton *et al.*, "The food contaminant deoxynivalenol, decreases intestinal barrier permeability and reduces claudin expression," *Toxicol. Appl. Pharmacol.*, 2009.

[107] Y. Luo *et al.*, "Toll-like receptor 5-mediated IL-17C expression in intestinal epithelial cells enhances epithelial host defense against F4+ ETEC infection," *Vet. Res.*, 2019.

[108] F. A. Omonijo *et al.*, "Thymol Improves Barrier Function and Attenuates Inflammatory Responses in Porcine Intestinal Epithelial Cells during Lipopolysaccharide (LPS)-Induced Inflammation," *J. Agric. Food Chem.*, 2019.

[109] Y. Wu *et al.*, “Protective effects of Lactobacillus plantarum on epithelial barrier disruption caused by enterotoxigenic Escherichia coli in intestinal porcine epithelial cells,” *Vet. Immunol. Immunopathol.*, 2016.

[110] H. Wang, N. Zhai, Y. Chen, C. Fu, and K. Huang, “OTA induces intestinal epithelial barrier dysfunction and tight junction disruption in IPEC-J2 cells through ROS/Ca2+-mediated MLCK activation,” *Environ. Pollut.*, vol. 242, pp. 106–112, 2018.

[111] A. K. Diesing *et al.*, “Vulnerability of polarised intestinal porcine epithelial cells to mycotoxin deoxynivalenol depends on the route of application,” *PLoS One*, 2011.

[112] C. Peng *et al.*, “β-Conglycinin-Induced Intestinal Porcine Epithelial Cell Damage via the Nuclear Factor κB/Mitogen-Activated Protein Kinase Signaling Pathway,” *J. Agric. Food Chem.*, 2019.

[113] Y. Zhao, G. Qin, R. Han, J. Wang, X. Zhang, and D. Liu, “β-conglycinin reduces the tight junction occludin and ZO-1 expression in IPEC-J2,” *Int. J. Mol. Sci.*, 2014.

[114] M. J. Gu *et al.*, “Barrier protection via Toll-like receptor 2 signaling in porcine intestinal epithelial cells damaged by deoxynivalnol,” *Vet. Res.*, 2016.

[115] H. Nakayama, N. Kitagawa, T. Otani, H. Iida, H. Anan, and T. Inai, “Ochratoxin A, citrinin and deoxynivalenol decrease claudin-2 expression in mouse rectum CMT93-II cells,” *Microscopy*, 2018.

[116] F. A. Omonijo *et al.*, “Thymol Improves Barrier Function and Attenuates Inflammatory Responses in Porcine Intestinal Epithelial Cells during Lipopolysaccharide (LPS)-Induced Inflammation,” *J. Agric. Food Chem.*, vol. 67, no. 2, pp. 615–624, Jan. 2019.

[117] H. Li, S. Zhuang, D. Li, J. Zhao, and Y. Ma, “Benign and malignant classification of mammogram images based on deep learning,” *Biomed. Signal Process. Control*, vol. 51, pp. 347–354, 2019.

[118] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012.

[119] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. 2016.

[120] Y. Xiao, J. Wu, Z. Lin, and X. Zhao, “A deep learning-based multi-model ensemble method for cancer prediction,” *Comput. Methods Programs Biomed.*, 2018.

[121] S. Yang, Z. Yin, Y. Wang, W. Zhang, Y. Wang, and J. Zhang, “Assessing cognitive mental workload via EEG signals and an ensemble deep learning classifier based on denoising autoencoders,” *Comput. Biol. Med.*, 2019.

[122] J. I. Orlando, E. Prokofyeva, M. del Fresno, and M. B. Blaschko, “An ensemble deep learning based approach for red lesion detection in fundus images,” *Comput. Methods Programs Biomed.*, 2018.

VIII. Acknowledgement

This dissertation would not have been possible without the help and support of a large number of people, and I thank for the opportunity and invaluable relationship I had with them.

First and foremost, I would like to sincerely thank my advisor Prof. Cheol-Heui Yun for providing me with an opportunity of participating in graduate studies and research under his invaluable guidance. I should thank him for his constant support and the flexibility. The more I work with Prof. Yun, the more fortunate I find myself. He gave me the courage and freedom to explore deep learning for information extraction at the very early days of the field. He provided me with valuable advice and suggestion whenever I need to deal with any challenges. It is his advice and enthusiasm that made me more determined to pursue my interests in my study and research.

I also wish to thank my committee members Prof. Byung-Chul Park and Prof. Tea-Sub Park for taking time out of their schedule and agreeing to serve in my committee and for their feedback on this thesis. Indeed, their support was very important toward the achievements during my Master's research.

I also would like to thank the Dept. of Agricultural Biotechnology, College of Agriculture and Life Sciences for providing all the resources and financial supports during my Master's program. I acknowledge that the excellent academical environment provided at the Seoul National University has been very influential and positively effective for me to progress in my Master's research.

I also appreciate the helps and collaborations of my colleagues and friends during my Master's studies. Specially, I should thank my lab-mates.

I'd like to thank my sister Sara Kassani who encouraged my interest in artificial inteligence, and helped me get my start in research.

Finally, to my parents and family, thank you for your ultimate sacrifices, shedding light on me even on the darkest days, and all the efforts you made to see me succeed.

Last, but not least, I would like to dedicate this thesis with everlasting love to my father's soul who is my biggest life inspiration.

IX. Appendix

Appendix 1. Source code for the evaluation of the results

```
import warnings
warnings.filterwarnings('always')
warnings.filterwarnings('ignore')
```

```
import sys
import math
import os
import numpy as np
from keras.utils.np_utils import to_categorical
from keras.optimizers import Adam, RMSprop,SGD
from keras.models import Model, Sequential
from keras.layers import Flatten, Dense, Activation, Dropout, GlobalAveragePooling2D, Input,concatenate
from keras.preprocessing.image import ImageDataGenerator
from keras.regularizers import l2, l1

import matplotlib.pyplot as plt

%matplotlib inline
```

```
Using TensorFlow backend.
```

```
import keras
import tensorflow as tf

print("Keras Version", keras.__version__)
print("tensorflow Version", tf.__version__)
# print("dim_ordering:", K.image_dim_ordering())
```

```
Keras Version 2.3.0
tensorflow Version 1.14.0
```

```
batch_size = 64
img_height, img_width = 256, 256
input_shape = (img_height, img_width, 3)
epochs = 400
```

```
train_dir = '../input/payam-thesis/train/'
test_dir = '../input/payam-thesis/test'
```

```
def preprocess_input(x):
    # 'RGB'->'BGR'
    x = x[:, :, ::-1]
    # Zero-center by imagenet mean pixel
    x[:, :, 0] -= 103.939
    x[:, :, 1] -= 116.779
    x[:, :, 2] -= 123.68
    return x
```

```
random_seed = np.random.seed(1142)

train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    preprocessing_function = preprocess_input,
    rotation_range=90,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.3,
    zoom_range=0.2,
    horizontal_flip=True,
    vertical_flip=True,
    fill_mode='reflect',
    validation_split= 0.2)

train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    seed = random_seed,
    shuffle = False,
    subset = 'training',
    class_mode='categorical')

validation_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    seed = random_seed,
    shuffle = False,
    subset = 'validation',
    class_mode='categorical')

test_datagen = ImageDataGenerator(rescale=1. / 255, preprocessing_function = preprocess_input)
test_generator = test_datagen.flow_from_directory(
    test_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    seed = random_seed,
    shuffle = False,
    class_mode='categorical')
```

```
Found 4611 images belonging to 3 classes.
Found 1152 images belonging to 3 classes.
Found 765 images belonging to 3 classes.
```

```
nb_train_samples = len(train_generator.filenames)
nb_validation_samples = len(validation_generator.filenames)
nb_test_samples = len(test_generator.filenames)

predict_size_train = int(math.ceil(nb_train_samples / batch_size))
predict_size_validation = int(math.ceil(nb_validation_samples / batch_size))
predict_size_test = int(math.ceil(nb_test_samples / batch_size))

num_classes = len(train_generator.class_indices)

print("nb_train_samples:", nb_train_samples)
print("nb_validation_samples:", nb_validation_samples)
print("nb_test_samples:", nb_test_samples)

print("\npredict_size_train:", predict_size_train)
print("predict_size_validation:", predict_size_validation)
print("predict_size_test:", predict_size_test)

print("\n num_classes:", num_classes)
```

```
nb_train_samples: 4611
nb_validation_samples: 1152
nb_test_samples: 765

predict_size_train: 73
predict_size_validation: 18
predict_size_test: 12

 num_classes: 3
```

```
vgg19_weights ="../input/full-keras-pretrained-no-top/vgg19_weights_tf_dim_ordering_tf_kernels_notop.h5"
inception_weights ="../input/full-keras-pretrained-no-top//inception_v3_weights_tf_dim_ordering_tf_kernels_notop.h5"
vgg16_weights ="../input/full-keras-pretrained-no-top/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5"
denseNet201_weights ="../input/full-keras-pretrained-no-top/densenet201_weights_tf_dim_ordering_tf_kernels_notop.h5"
denseNet121_weights ="../input/full-keras-pretrained-no-top/densenet121_weights_tf_dim_ordering_tf_kernels_notop.h5"
resenet50_weights ="../input/full-keras-pretrained-no-top/resnet50_weights_tf_dim_ordering_tf_kernels_notop.h5"
inception_resnet_v2_weights ="../input/full-keras-pretrained-no-top/inception_resnet_v2_weights_tf_dim_ordering_tf_kernels
nasnet_weights ="../input/full-keras-pretrained-no-top/nasnet_large_no_top.h5"
nasnet_mobile_weights ="../input/full-keras-pretrained-no-top/nasnet_mobile_no_top.h5"
mobilenet_weights ="../input/full-keras-pretrained-no-top/mobilenet_1_0_224_tf_no_top.h5"
```

```
from keras.applications.vgg19 import VGG19
from keras.applications.inception_v3 import InceptionV3
from keras.applications.vgg16 import VGG16
from keras.applications.xception import Xception, preprocess_input
from keras.applications import DenseNet201
from keras.applications import DenseNet121
from keras.applications import ResNet50
from keras.applications.inception_resnet_v2 import InceptionResNetV2
from keras.applications import NASNetLarge, NASNetMobile
from keras.applications import MobileNet
```

```
input_tensor = Input(shape = input_shape)

base_model1=InceptionV3(input_shape= input_shape,weights=inception_weights, include_top=False, input_tensor=input_tensor)
base_model2=DenseNet201(input_shape= input_shape,weights=denseNet201_weights, include_top=False,
                        input_tensor=input_tensor)

x1 = base_model1.output
x1 = GlobalAveragePooling2D()(x1)

x2 = base_model2.output
x2 = GlobalAveragePooling2D()(x2)

merge = concatenate([x1, x2])
predictions = Dense(num_classes, activation='softmax')(merge)

model = Model(inputs=input_tensor,outputs=predictions)
```

```
for i, layer in enumerate(model.layers):
    print(i, layer.name)
```

```
0 input_1
1 zero_padding2d_1
2 conv1/conv
3 conv1/bn
4 conv1/relu
5 zero_padding2d_2
6 pool1
7 conv2_block1_0_bn
8 conv2_block1_0_relu
9 conv2_block1_1_conv
10 conv2_block1_1_bn
11 conv2_block1_1_relu
12 conv2_block1_2_conv
13 conv2_block1_concat
14 conv2_block2_0_bn
15 conv2_block2_0_relu
16 conv2_block2_1_conv
17 conv2_block2_1_bn
18 conv2_block2_1_relu
```

```
c1 = model.layers[585].output
c1 = GlobalAveragePooling2D()(c1)

c2 = model.layers[601].output
c2 = GlobalAveragePooling2D()(c2)

c3 = model.layers[634].output
c3 = GlobalAveragePooling2D()(c3)

c4 = model.layers[667].output
c4 = GlobalAveragePooling2D()(c4)

c5 = model.layers[708].output
c5 = GlobalAveragePooling2D()(c5)

c6 = model.layers[739].output
c6 = GlobalAveragePooling2D()(c6)

c7 = model.layers[787].output
c7 = GlobalAveragePooling2D()(c7)

c8 = model.layers[835].output
c8 = GlobalAveragePooling2D()(c8)

c9 = model.layers[883].output
c9 = GlobalAveragePooling2D()(c9)

c10 = model.layers[933].output
c10 = GlobalAveragePooling2D()(c10)

c11 = model.layers[954].output
c11 = GlobalAveragePooling2D()(c11)

c12 = model.layers[996].output
c12 = GlobalAveragePooling2D()(c12)

con = concatenate([c1, c2, c3, c4, c5, c6, c7, c8, c9, c10, c11, c12])

bottleneck_final_model = Model(inputs=model.input, outputs=con)
```

```
os.mkdir("extracted_features")
extracted_features_dir = "extracted_features/"
model_name = "densenet201_InceptionResNetV2_descriptors"
```

```
bottleneck_features_train = bottleneck_final_model.predict_generator(train_generator, predict_size_train)
np.save(extracted_features_dir+'bottleneck_features_train_'+model_name+'.npy', bottleneck_features_train)
```

```
bottleneck_features_validation = bottleneck_final_model.predict_generator(validation_generator, predict_size_validation)
np.save(extracted_features_dir+'bottleneck_features_validation_'+model_name+'.npy', bottleneck_features_validation)
```

```
bottleneck_features_test = bottleneck_final_model.predict_generator(test_generator, predict_size_test)
np.save(extracted_features_dir+'bottleneck_features_test_'+model_name+'.npy', bottleneck_features_test)
```

```
train_data = np.load(extracted_features_dir+'bottleneck_features_train_'+model_name+'.npy')
validation_data = np.load(extracted_features_dir+'bottleneck_features_validation_'+model_name+'.npy')
test_data = np.load(extracted_features_dir+'bottleneck_features_test_'+model_name+'.npy')

train_labels = train_generator.classes
train_labels = to_categorical(train_labels, num_classes=num_classes)

validation_labels = validation_generator.classes
validation_labels = to_categorical(validation_labels, num_classes=num_classes)

test_labels = test_generator.classes
test_labels = to_categorical(test_labels, num_classes=num_classes)
```

```
model = Sequential()
model.add(Dense(4096, activation='tanh',kernel_regularizer=l2(1e-05),  bias_regularizer=l2(1e-06),
                activity_regularizer=l1(0.001)))
model.add(Dense(128, activation='tanh',kernel_regularizer=l2(1e-05),  bias_regularizer=l2(1e-06),
                activity_regularizer=l1(0.001)))
model.add(Dense(64, activation='tanh',kernel_regularizer=l2(1e-05),  bias_regularizer=l2(1e-06),
                activity_regularizer=l1(0.001)))
model.add(Dropout(0.5))
model.add(Dense(num_classes, activation='softmax'))

adam_opt=Adam(lr = 0.0001, beta_1=0.6, beta_2=0.8)
model.compile(optimizer=adam_opt, loss='categorical_crossentropy', metrics=['accuracy'])
history = model.fit(train_data, train_labels,
                    epochs=epochs,
                    batch_size=batch_size,
                    validation_data=(validation_data, validation_labels),
                    verbose= 2)
```

```
Train on 4611 samples, validate on 1152 samples
Epoch 1/400
 - 3s - loss: 31.6564 - accuracy: 0.5532 - val_loss: 22.2448 - val_accuracy: 0.4965
Epoch 2/400
 - 1s - loss: 16.6631 - accuracy: 0.6864 - val_loss: 17.6401 - val_accuracy: 0.4479
Epoch 3/400
 - 1s - loss: 13.8761 - accuracy: 0.7165 - val_loss: 16.0890 - val_accuracy: 0.4618
Epoch 4/400
 - 1s - loss: 12.5201 - accuracy: 0.7482 - val_loss: 15.3052 - val_accuracy: 0.7682
Epoch 5/400
 - 1s - loss: 11.6471 - accuracy: 0.7482 - val_loss: 14.7140 - val_accuracy: 0.3316
Epoch 6/400
 - 1s - loss: 11.0217 - accuracy: 0.7482 - val_loss: 12.9061 - val_accuracy: 0.6476
Epoch 7/400
 - 1s - loss: 10.5064 - accuracy: 0.7805 - val_loss: 13.7485 - val_accuracy: 0.7075
Epoch 8/400
 - 1s - loss: 10.1385 - accuracy: 0.7849 - val_loss: 13.4153 - val_accuracy: 0.6024
Epoch 9/400
 - 1s - loss: 9.8014 - accuracy: 0.7781 - val_loss: 12.1209 - val_accuracy: 0.9288
```

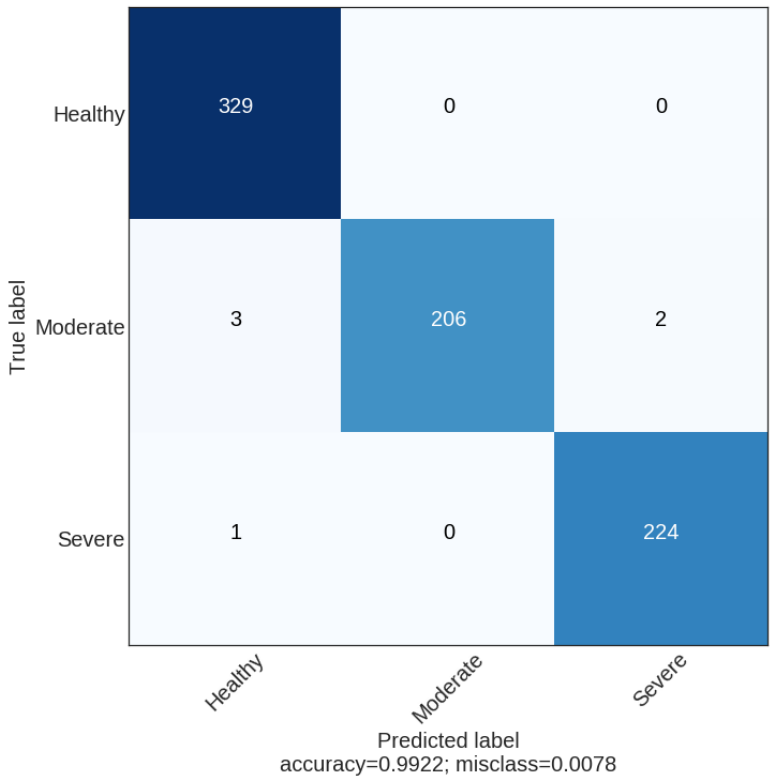
```
(eval_loss, eval_accuracy) = model.evaluate(validation_data, validation_labels, batch_size= batch_size, verbose=1)

print("Validation Accuracy: {:.4f}%".format(eval_accuracy * 100))
print("Validation Loss: {}".format(eval_loss))
```

```
1152/1152 [==============================] - 0s 63us/step
Validation Accuracy: 98.6111%
Validation Loss: 7.581387069490221
```

```
from sklearn.metrics import accuracy_score,roc_curve, confusion_matrix, roc_auc_score, auc, f1_score

preds = model.predict(test_data)

predictions = [i.argmax() for i in preds]
y_true = [i.argmax() for i in test_labels]
cm = confusion_matrix(y_pred=predictions, y_true=y_true)

print('Accuracy {}'.format(accuracy_score(y_true=y_true, y_pred=predictions)))
```

```
Accuracy 0.9921568627450981
```

```
plt.rcParams["axes.grid"] = False
plt.rcParams.update({'font.size': 20})

labels = []

label = test_generator.class_indices
indexlabel = dict((value, key) for key, value in label.items())

for k,v in indexlabel.items():
    labels.append(v)

from sklearn.metrics import confusion_matrix
import itertools
def plot_confusion_matrix(cm, classes, normalize=False, title='Confusion matrix', cmap=plt.cm.Blues):
    accuracy = np.trace(cm) / float(np.sum(cm))
    misclass = 1 - accuracy
    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        print("Normalized confusion matrix")
    else:
        print('Confusion Matrix')
    print(cm)
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    fmt = '.2f' if normalize else 'd'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")

    plt.tight_layout()
    plt.ylabel('True label')
    plt.xlabel('Predicted label\naccuracy={:0.4f}; misclass={:0.4f}'.format(accuracy, misclass))

plt.figure(figsize=(10,10))
plot_confusion_matrix(cm, classes=labels, title=' ')
```



```
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import confusion_matrix, classification_report
y_pred=predictions
y_pred_probabilities=y_pred

# y_pred = np.argmax(y_pred,axis = 1)
y_actual = y_true

classnames=[]
for classname in test_generator.class_indices:
    classnames.append(classname)

confusion_mtx = confusion_matrix(y_actual, y_pred)
print(confusion_mtx)
target_names = classnames
print(classification_report(y_actual, y_pred, target_names=target_names))
```

```
[[329   0   0]
 [  3 206   2]
 [  1   0 224]]
              precision    recall  f1-score   support

     Healthy       0.99      1.00      0.99       329
    Moderate       1.00      0.98      0.99       211
      Severe       0.99      1.00      0.99       225

    accuracy                           0.99       765
   macro avg       0.99      0.99      0.99       765
weighted avg       0.99      0.99      0.99       765
```

```
total=sum(sum(cm))

sensitivity = cm[0,0]/(cm[0,0]+cm[1,0])
print('Sensitivity : ', sensitivity*100 )

Specificity = cm[1,1]/(cm[1,1]+cm[0,1])
print('Specificity : ', Specificity*100 )
```

```
Sensitivity :  99.09638554216868
Specificity :  100.0
```

```
y_pred_class = model.predict(test_data, verbose=1)

y_pred_class = [np.argmax(r) for r in y_pred_class]
test_y = [np.argmax(r) for r in test_labels]


from sklearn.metrics import confusion_matrix, precision_score, recall_score, f1_score, roc_auc_score, roc_curve

# Precision
print('Precision = ', precision_score(test_y, y_pred_class, average='weighted'))
# (None, 'micro', 'macro', 'weighted', 'samples')

# Recall
print('Recall = ', recall_score(test_y, y_pred_class, average='weighted'))

# f1_score
print('f1_score = ', f1_score(test_y, y_pred_class, average='weighted'))
```

```
765/765 [==============================] - 0s 69us/step
Precision =  0.9922312387000917
Recall =  0.9921568627450981
f1_score =  0.992137822895831
```

```
from sklearn.metrics import roc_auc_score
from sklearn.preprocessing import LabelBinarizer

def multiclass_roc_auc_score(y_test, y_pred, average="weighted"):
    label_binarizer = LabelBinarizer()
    label_binarizer.fit(y_test)

    truth = label_binarizer.transform(y_test)
    pred = label_binarizer.transform(y_pred)
    return roc_auc_score(truth, pred, average=average)
# roc_auc_score
print('roc_auc_score = ', multiclass_roc_auc_score(test_y, y_pred_class))
```

```
roc_auc_score =  0.9935609921848455
```

Appendix 2. Copyright and permission

1. "[Fig. 4. Fig. 5.] Reused from Mediators of Inflammation, Volume 2017:2748192, M. Kern, D. Günzel, J. R. Aschenbach, K. Tedin, A. Bondzio and U. Lodemann, Altered Cytokine Expression and Barrier Properties after In Vitro Infection of Porcine Epithelial Cells with Enterotoxigenic Escherichia coli and Probiotic Enterococcus faecium, Copyright (2017), with permission under the Creative Commons Attribution License."
https://creativecommons.org/licenses/by/4.0/

2. "[Fig. 1B. Fig. 2F] Reused from Asian-Australasian Journal of Animal Sciences, Volume 27, No.4, Min Jeong Gu, Sun Kwang Song, Sung Moo Park, In Kyu Lee, and Cheol-Heui Yun, Bacillus subtilis Protects Porcine Intestinal Barrier from Deoxynivalenol via Improved Zonula Occludens-1 Expression, Pages No. 580-586, Copyright (2014), with permission under the Creative Commons Attribution License. " https://creativecommons.org/licenses/by-nc/3.0/

3. "[Fig. 3C. Fig. 6C.] Reused from International Journal of Molecular Sciences, Volume 19(7), Zhenya Zhai, Xiaojun Ni, Chenglong Jin, Wenkai Ren, Jie Li, Jinping Deng, Baichuan Deng and Yulong Yin, Cecropin A Modulates Tight Junction-Related Protein Expression and Enhances the Barrier Function of Porcine Intestinal Epithelial Cells by Suppressing the MEK/ERK Pathway, Copyright (2018), with permission under the Creative Commons Attribution License."
https://creativecommons.org/licenses/by/4.0/

4. "[Fig. 7] Reused from Physiological Reports, Volume 3, No.4, Hao-Yu Liu, Stefan Roos, Hans Jonsson, David Ahl, Johan Dicksved, Jan Erik Lindberg, and Torbjörn Lundh, Effects of Lactobacillus johnsonii and Lactobacillus reuteri on gut barrier function and heat shock proteins in intestinal porcine epithelial cells, Copyright (2015), with permission under the Creative Commons Attribution License." https://creativecommons.org/licenses/by/4.0/

5. "[Fig. 4. c, Fig. 8.e,] Reused from Toxicology Letters, Volume 311, Chen Ying,Wang Hong,Zhai Nianhui,Wang Chunlei,Huang Kehe,Pan Cuiling, Nontoxic concentrations of OTA aggravate DON-induced intestinal barrier dysfunction in IPEC-J2 cells via activation of NF-κB signaling pathway, Pages No. 114-124, Copyright (2019), with permission from Elsevier."
https://s100.copyright.com/CustomerAdmin/PLF.jsp?ref=ca77b09e-cf5a-48c6-9b2b-9a4dfff8d5cd

6. "[Fig. 3.] Reused from Toxicology and Applied Pharmacology, Volume 237, Issue 1,
Philippe Pinton,Jean-Philippe Nougayrède,Juan-Carlos Del Rio,Carolina Moreno,Daniela E. Marin,Laurent Ferrier,Ana-Paula Bracarense,Martine Kolf-Clauw,Isabelle P. Oswald, The food contaminant deoxynivalenol, decreases intestinal barrier permeability and reduces claudin expression, Pages No. 41-48, Copyright (2019), with permission from Elsevier."
https://s100.copyright.com/CustomerAdmin/PLF.jsp?ref=bb52ebb5-3282-4b90-96b7-06ede8435616

7. "[Fig. 4C. Fig. 6A.] Reused from Veterinary Research, Volume 50, No.48, Yu Luo, Jia Xu, Chaoying Zhang, Chunyan Jiang, Yanfeng Ma, Haijian He, Yuan Wu, Bert Devriendt, Eric Cox, and Hongbin Zhang, Toll-like receptor 5-mediated IL-17C expression in in7testinal epithelial cells enhances epithelial host defense against F4+ ETEC infection, Copyright (2018), with permission under the Creative Commons Attribution License." https://creativecommons.org/licenses/by/4.0/

8. "[Fig. 7.] Reused from Journal of Agricultural and Food Chemistry, Volume 67, Faith A. Omonijo, Shangxi Liu, Qianru Hui, Hua Zhang, Ludovic Lahaye, Jean-Christophe BodinJoshua Gong, Martin Nyachoti and Chengbo Yang, Thymol Improves Barrier Function and Attenuates Inflammatory Responses in Porcine Intestinal Epithelial Cells during Lipopolysaccharide (LPS)-Induced Inflammation, Pages No. 615-624, Copyright (2018), with permission under the American Chemical Society License."
https://s100.copyright.com/AppDispatchServlet#formTop

9. "[Fig. 4.] Reused from Veterinary Immunology and Immunopathology, Volume 172, Yunpeng Wu, Cui Zhu, Zhuang Chen, Zhongjian Chen, Weina Zhang, Xianyong Ma, Li Wang, Xuefen Yang, Zongyong Jiang, Protective effects of Lactobacillus plantarum on epithelial barrier disruption caused by enterotoxigenic Escherichia coli in intestinal porcine epithelial cells, Pages No. 55-63, Copyright (2016), with permission from Elsevier." https://s100.copyright.com/CustomerAdmin/PLF.jsp?ref=80a6c475-17c5-49e0-98c5-5b663672556f

10. "[Fig. 1.C,D, Fig. 2.F,N, Fig. 3.D, Fig. 4.F,E, Fig. 5.G,] Reused from Environmental Pollution, Volume 242, Part A, Hong Wang,Nianhui Zhai,Ying Chen,Chongyang Fu,Kehe Huang, OTA induces intestinal epithelial barrier dysfunction and tight junction disruption in IPEC-J2 cells through ROS/Ca2+-mediated MLCK activation, Pages No. 106-112, Copyright (2018), with permission from Elsevier." https://s100.copyright.com/CustomerAdmin/PLF.jsp?ref=a7347889-c164-4aec-8d95-12a6a732a09f

11. "[Fig. 7. Fig. 8] Reused from PloS one, Volume 6, Issue 2, Anne-Kathrin Diesing, Constanze Nossol, Sven Dänicke, Nicole Walk, Andreas Post, Stefan Kahlert, Hermann-Josef Rothkötter, Jeannette Kluess, Vulnerability of Polarised Intestinal Porcine Epithelial Cells to Mycotoxin Deoxynivalenol Depends on the Route of Application, Copyright (2011), with permission under the Creative Commons Attribution License." https://creativecommons.org/licenses/by/4.0/

12. "[Fig. 7. Fig. 8. Fig. 8] Reused from Journal of Agricultural and Food Chemistry, Volume 67, Chenglu Peng, Xuedong Ding, Lei Zhu, Mengchu He, Yingshuang Shu, Yu Zhang,Yu Li, Xichun Wang, Shibin Feng, Jinchun Li, Jinjie Wu, β Conglycinin-Induced Intestinal Porcine Epithelial Cell Damage via the Nuclear Factor κB/Mitogen-Activated Protein Kinase Signaling Pathway, Pages No. 9009−9021, Copyright (2019), with permission under the American Chemical Society License." https://s100.copyright.com/AppDispatchServlet

13. "[Fig. 4] Reused from International Journal of Molecular Sciences, Volume 15, Issue 2, Yuan Zhao, Guixin Qin, Rui Han, Jun Wang, Xiaodong Zhang, Dandan Liu, β-Conglycinin Reduces the Tight Junction Occludin and ZO-1Expression in IPEC-J2, Pages No. 1915–1926, Copyright (2014), with permission under the Creative Commons Attribution License." https://creativecommons.org/licenses/by-nc-sa/3.0/

14. "[Fig. 7] Reused from Veterinary Research, 47: 25, Min Jeong Gu, Sun Kwang Song, In Kyu Lee, Seongyeol Ko, Seung Eun Han, Suhan Bae, Sang Yun Ji, Byung-Chul Park, Ki-Duk Song, Hak-Kyo Lee, Seung Hyun Han, Cheol-Heui Yun, Barrier protection via Toll-like receptor 2 signaling in porcine intestinal epithelial cells damaged by deoxynivalnol, Copyright (2016), with permission under the Creative Commons Attribution License." https://creativecommons.org/licenses/by/4.0/