International Journal of Academic Engineering Research (IJAER)
ISSN: 2643-9085
Vol. 6 Issue 3, March - 2022, Pages:15-28

Classification of Anomalies in Gastrointestinal Tract Using Deep Learning

Ibtesam M. Dheir and Samy S. Abu Naser

Deprtment of Information Technology,
Faculty of Engineering and Information Technology,
Al-Azhar University, Gaza, Palestine

Abstract*: Automatic detection of diseases and anatomical landmarks in medical images by the use of computers is important and considered a challenging process that could help medical diagnosis and reduce the cost and time of investigational procedures and refine health care systems all over the world. Recently, gastrointestinal (GI) tract disease diagnosis through endoscopic image classification is an active research area in the biomedical field. Several GI tract disease classification methods based on image processing and machine learning techniques have been proposed by diverse research groups in the recent past. However, yet effective and comprehensive deep ensemble neural network-based classification model with high accuracy classification results is not available in the literature. In this thesis, we review ways and mechanisms to use deep learning techniques to research on multi-disease computer-aided detection about gastrointestinal and identify these images. We re-trained five state-of-the-art neural network architectures, VGG16, ResNet, MobileNet, Inception-v3, and Xception on the Kvasir dataset to classify eight categories that include an anatomical landmark (pylorus, z-line, cecum), a diseased state (esophagitis, ulcerative colitis, polyps), or a medical procedure (dyed lifted polyps, dyed resection margins) in the Gastrointestinal Tract. Our models have showed results with a promising accuracy which is a remarkable performance with respect to the state-of-the-art approaches. The resulting accuracies achieved using VGG, ResNet, MobileNet, Inception-v3, and Xception were 98.3%, 92.3%, 97.6%, 90% and 98.2%, respectively. As it appears, the most accurate result has been achieved when retraining VGG16 and Xception neural networks with accuracy reache to 98% due to its high performance on training on ImageNet dataset and internal structure that support classification problems.*

Keywords: Deep Learning, Machine Learning, Transfer Learning, Artificial Intelligence, neural network, Gastrointestinal Tract.

1. Introduction

Machine learning (ML) is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed [1]. ML is the study of computer algorithms that improve automatically through experience. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as image recognition, Medical Diagnosis, and classification, where it is difficult or infeasible to develop conventional algorithms to perform the needed tasks [2].

Deep learning (also known as deep structured learning or hierarchical learning) is an artificial intelligence function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence (AI) that has networks capable of learning unsupervised from data that is unstructured or unlabeled. It is also known as deep neural learning or deep neural network [3].

Deep learning, a division of machine learning, uses a hierarchical level of artificial neural networks to perform the process of machine learning. The artificial neural networks are constructed like the human brain, with neuron nodes linked together like a web [4].

In deep learning, each level learns to transform its input data into a slightly more abstract and composite representation. In an image recognition application, the raw input may be a matrix of pixels; the first representational layer may abstract the pixels and encode edges; the second layer may compose and encode arrangements of edges; the third layer may encode a nose and eyes; and the fourth layer may recognize that the image contains a face. Importantly, a deep learning process can learn which features to optimally place in which level on its own. (Of course, this does not completely obviate the need for hand-tuning; for example, varying numbers of layers and layer sizes can provide different degrees of abstraction)[5,6].

2. The Gastrointestinal tract

The digestive system is made up of the gastrointestinal tract (fig. (2.1))—also called the GI tract or digestive tract—and the liver, pancreas, and gallbladder. The GI tract is a series of hollow organs joined in a long, twisting tube from the mouth to the anus. The hollow organs that make up the GI tract are the mouth, esophagus, stomach, small intestine, large intestine, and anus. The liver, pancreas, and gallbladder are the solid organs of the digestive system [7].

The small intestine has three parts. The first part is called the duodenum. The jejunum is in the middle and the ileum is at the end. The large intestine includes the appendix, cecum, colon, and rectum. The appendix is a finger-shaped pouch attached to the cecum. The cecum is the first part of the large intestine. The colon is next. The rectum is the end of the large intestine [7].

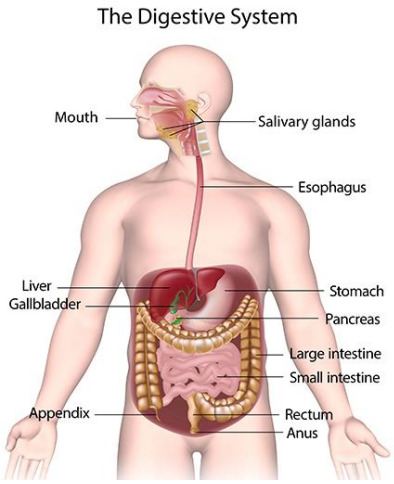


Fig. (2.1): The Gastrointestinal tract [7]

The human digestive system may be affected by several diseases. As an example, three of the eight most common cancers overall are located in the gastrointestinal (GI) tract. Altogether accounts for about 2.8 million new cases and 1.8 million deaths per year [8].

Endoscopic examinations are procedures using an endoscope to diagnose or treat a condition which are the gold standards for the investigation of the GI tract. There are several types of endoscopy. Gastroscopy which is shown in fig. (2.2) is an examination of the upper GI tract including the esophagus, stomach, and first part of the small bowel using an endoscope which is shown in fig. (2.4) — a long, thin, flexible tube containing a camera and a light — to view the lining of these organs, while colonoscopy which is shown in fig. (2.3) is a common, safe test to examine the lining of the large bowel(colon) and rectum. Both these examinations are real-time video examinations of the inside of the GI tract by the use of digital high definition endoscopes. Endoscopic examinations are resource-demanding and require both expensive technical equipment and trained personnel.

Fig. (2.2): Gastroscopy[8]

Fig. (2.3): Colonoscopy[8]

Fig. (2.4): endoscope[8]

2.1 Anatomical Landmarks

An anatomical landmark is a recognizable feature within the GI tract that is easily visible through the endoscope. They are essential for navigating and as a reference point to describe the location of a given finding. The landmarks may also be typical sites for pathology like ulcers or inflammation. A complete endoscopic rapport should preferably contain both brief descriptions and image documentation of the most important anatomical landmarks [9]. The three anatomical landmark we dealt with are pylorus, zline, and cecum.

2.1.1 Z-line

The Z-line marks the transition site between the esophagus and the stomach. Endoscopically, it is visible as a clear border where the white mucosa in the esophagus meets the red gastric mucosa. Recognition and assessment of the Z-line are important in order to determine whether the disease is present or not. For example, this is the area where signs of gastro esophageal reflux may appear. The Z-line is also useful as a reference point when describing pathology in the esophagus [10]. An example of the Z-line is shown in fig. (2.5).

Fig. (2.5): Z-line[56]

2.1.2 Cecum

The cecum, pouch, or large tubelike structure in the lower abdominal cavity that receives undigested food material from the small intestine. And is considered the first region of the large intestine. It is separated from the ileum (the final portion of the small intestine) by the ileocecal valve (also called Bauhin valve), which limits the rate of food passage into the cecum and may help prevent material from returning to the small intestine. The main functions of the cecum are to absorb fluids and salts that remain after completion of intestinal digestion and absorption and to mix its contents with a lubricating substance, mucus[11]. Reaching the cecum is the proof for a complete colonoscopy and the completion rate has shown to be a valid quality indicator for colonoscopy [12]. Therefore, recognition and documentation of the cecum are important. One of the characteristics hallmarks of the cecum is the appendiceal orifice. Fig. (2.6) shows an example of the appendiceal orifice visible as a crescent-shaped slit, and the green picture in figure 6 shows the scope configuration for the cecal position [10].

Fig. (2.6): Cecum [56]

2.1.3 Pylorus

The pylorus is defined as the area around the opening from the stomach into the first part of the small bowel (duodenum) . The opening contains circumferential muscles that regulates the movement of food from the stomach. The main functions of the pylorus are to prevent intestinal contents from reentering the stomach when the small intestine contracts and to limit the passage of large food particles or undigested material into the intestine. The term pylorus is used to refer to the pyloric sphincter and can also be applied to that portion of the stomach immediately above the pyloric sphincter. An image of a normal pylorus viewed from inside the stomach by an endoscopic is shown in fig. (2.7). The identification of pylorus is necessary for endoscopic instrumentation to the duodenum, one of the challenging maneuvers within gastroscopy. A complete gastroscopy includes inspection on both sides of the pyloric opening to reveal findings like ulcerations, erosions or stenosis. Here, the smooth, round opening is visible as a dark circle surrounded by homogeneous pink stomach mucosa [13] [10].

Fig. (2.1): pylorus [56]

2.2 Pathological findings

A pathological finding in this context is an abnormal feature within the gastrointestinal tract. Endoscopically, it is visible as damage or change in the normal mucosa. The finding may be signs of an ongoing disease or a precursor to for example cancer. Detection and classification of pathology are important in order to initiate correct treatment and/or follow-up of the patient.

2.2.1 Esophagitis

Esophagitis is any inflammation or irritation of the esophagus and it is visible as a break in the esophageal mucosa in relation to the Z-line. The esophagus is the tube that sends food from mouth to stomach. Fig. (2.8) shows an example with red mucosal tongues projecting up in the white esophageal lining. The grade of inflammation is defined by the length of the mucosal breaks and the proportion of the circumference involved [10]. Common causes include acid

reflux, side effects of certain medications, and bacterial or viral infections. Reflux is when the stomach contents and acids back up into the esophagus [14]. Clinically, detection is necessary for treatment initiation to relieve symptoms and prevent further development of possible complications.

Fig. (2.8): Esophagitis [56]

2.2.2 Polyps

Polyps are abnormal tissue growths that most often look like small, flat bumps or tiny mushroomlike stalks. Most polyps are small and less than half an inch wide [15]. Polyps in GI are lesions within the bowel detectable as mucosal outgrows. Figure 7 shows an example of a typical polyp. The polyps are flat, elevated or pedunculated, and can be distinguished from normal mucosa by color and surface pattern. Most bowel polyps are harmless, but some have the potential to grow into cancer. Detection and removal of polyps are therefore important to prevent development of colorectal cancer. Since polyps may be overlooked by the doctors, automatic detection would most likely improve examination quality. The green boxes within the fig. (2.9) show an illustration of the endoscope configuration [10]. In live endoscopy, this helps to determine the current localization of the endoscope-tip (and thereby also the polyp site) within the length of the bowel. Automatic computer-aided detection of polyps would be valuable both for diagnosis, assessment, and reporting.

Fig. (2.9): Polyps [56]

2.2.3 Ulcerative colitis

Ulcerative colitis is a chronic inflammatory disease affecting the large bowel. It causes irritation, inflammation, and ulcers in the lining of your large intestine. The disease may have a large impact on the quality of life, and diagnosis is mainly based on colonoscopic findings. The degree of inflammation varies from none, mild, moderate, and severe, all with different endoscopic aspects. For example, in mild disease, the mucosa appears swollen and red, while in moderate cases, ulcerations are prominent. Fig. (2.10) shows an example of ulcerative colitis with bleeding, swelling, and ulceration of the mucosa. The white coating visible in the illustration is fibrin covering the wounds [16][10].

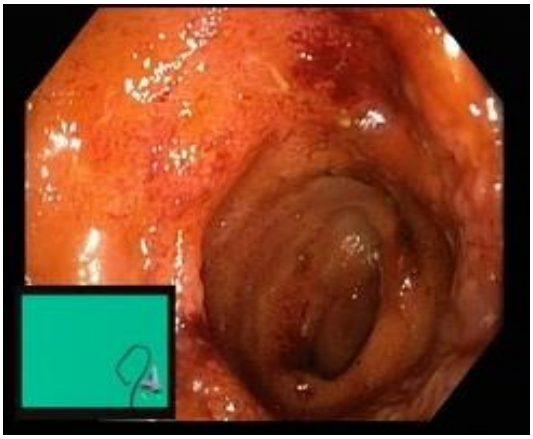


Fig. (2.10): Ulcerative colitis [56]

2.3 Polyp removal

Polyps in the large bowel may be precursors of cancer and are therefore removed during endoscopy if possible. One of the polyp removal techniques is called endoscopic mucosal resection (EMR). This includes injection of a liquid underneath the polyp, lifting the polyp from the underlying tissue. The polyp is then captured and removed by use of a snare. The lifting minimizes risk of mechanical or electrocautery damage to the deeper layers of the GI wall. Staining dye (i.e., diluted indigo carmine) is added to facilitate accurate identification of the polyp margins [17].

2.3.1 Dyed and Lifted Polyps.

The light blue polyp shown in figure 9 margins is clearly visible against the darker normal mucosa(Fig. (2.11) shows an example of a polyp lifted by injection of saline and indigo carmine.) . Additional valuable information related to automatic reporting may involve successfulness of the lifting and eventual presence of non-lifted areas that might indicate malignancy [10].

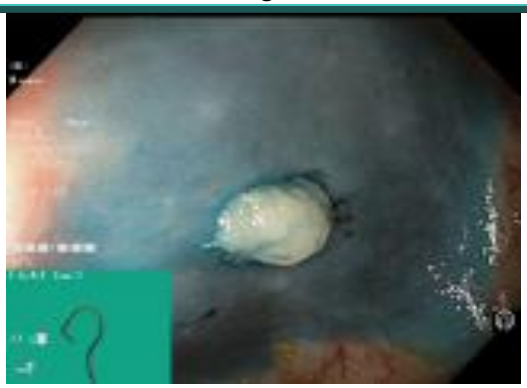


Fig. (2.11): Dyed and Lifted Polyps [56]

2.3.2 Dyed Resection Margins.

The resection margins are important in order to evaluate whether the polyp is completely removed or not. Residual polyp tissue may lead to continued growth and in worst case malignancy development [10]. Fig. (2.12) illustrates the resection site after removal of a polyp.

Fig. (2.12) : Dyed Resection Margins [56]

3. Convolutional Neural Network

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which are able to perform relatively complex tasks with images, sounds, texts, videos, etc. It is a multi-layer neural network designed to analyze visual inputs and perform tasks such as image classification, segmentation and object detection, which can be useful for autonomous vehicles. [18] It assigns learnable weights and biases to various objects in image and be able to differentiate one from the other. The first successful convolution networks were developed in the late 1990s by Professor Yann LeCunn for Bell Labs[18].

Convolutional Neural Networks are very similar to ordinary Neural Networks, they are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. The whole network still expresses a single differentiable score function: from the raw image pixels on one end to class scores at the other. And they still have a loss function (e.g. SVM/Softmax) on the last (fully-connected) layer and all the tips/tricks we developed for learning regular Neural Networks still apply.

CNN uses a variation of multilayer perceptions designed to require minimal preprocessing [19]. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics [20,21].

In computer vision, CNNs have been known to be powerful visual models that yield hierarchies of features enabling accurate segmentation. They are also known to perform predictions relatively faster than other algorithms while maintaining competitive performance at the same time [18].

A simple ConvNet is a sequence of layers, and every layer of a ConvNet transforms one volume of activations to another through a differentiable function. Three main types of layers used to build ConvNet architectures: Convolutional Layer, Pooling Layer, and Fully-Connected Layer (exactly as seen in regular Neural Networks). We will stack these layers to form a full ConvNet architecture.

3.1 Convolutional Layer

When programming a CNN, the input is a tensor with shape (number of images) x (image width) x (image height) x (image depth). Then after passing through a convolutional layer, the image becomes abstracted to a feature map, with shape (number of images) x (feature map width) x (feature map height) x (feature map channels). A convolutional layer within a neural network should have the following attributes:

- Convolutional kernels defined by a width and height (hyper-parameters).
- The number of input channels and output channels (hyper-parameter).
- The depth of the Convolution filter (the input channels) must be equal to the number channels (depth) of the input feature map.

Convolutional layers convolve the input and pass its result to the next layer. This is similar to the response of a neuron in the visual cortex to a specific stimulus.[22]. Each convolutional neuron processes data only for its receptive field. Although fully connected feedforward neural networks can be used to learn features as well as classify data, it is not practical to apply this architecture to images. A very high number of neurons would be necessary, even in shallow (opposite of deep) architecture, due to the very large input sizes associated with images, where each pixel is a relevant variable. For instance, a fully connected layer for a (small) image of size 100 x 100 has 10,000 weights for each neuron in the second layer. The convolution operation brings a

solution to this problem as it reduces the number of free parameters, allowing the network to be deeper with fewer parameters.[24] For instance, regardless of image size, tiling regions of size 5 x 5, each with the same shared weights, requires only 25 learnable parameters. In this way, it resolves the vanishing or exploding gradients problem in training traditional multi-layer neural networks with many layers by using backpropagation[23].

3.2 Pooling Layer

Convolutional networks may include local or global pooling layers to streamline the underlying computation. Pooling layers reduce the dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. Local pooling combines small clusters, typically 2 x 2. Global pooling acts on all the neurons of the convolutional layer.[25,26] In addition, pooling may compute a max or an average. Max pooling uses the maximum value from each of a cluster of neurons at the prior layer [27,28]. Average pooling uses the average value from each of a cluster of neurons at the prior layer [29].

3.3 Rectified Linear Unit Layer

The Rectified Linear Unit is the most commonly deployed activation function for the output in deep learning models. The function returns 0 if it receives any negative input, but for any positive value xx it returns that value back. So it can be written as f(x)=max(0,x).
Graphically it looks like this in Fig. (2.13):

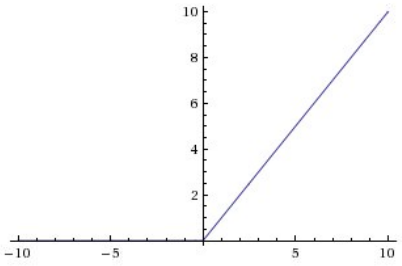


Fig. (2.13): Rectified Linear Unit Layer

3.4 Dropout Layer

Dropout is a regularization method that approximates training a large number of neural networks with different architectures in parallel. During training, some number of layer outputs are randomly ignored or "*dropped out*." This has the effect of making the layer look-like and be treated-like a layer with a different number of nodes and connectivity to the prior layer. In effect, each update to a layer during training is performed with a different "*view*" of the configured layer. Dropout has the effect of making the training process noisy, forcing nodes within a layer to probabilistically take on more or less responsibility for the inputs [30].

3.5 Batch Normalization Layer

Batch normalization is a technique for training very deep neural networks that standardizes the inputs to a layer for each mini-batch. This has the effect of stabilizing the learning process and dramatically reducing the number of training epochs required to train deep networks [31]. Batch normalization allows each layer of a network to learn by itself a little bit more independently of other layers [32]. It reduces overfitting because it has a slight regularization effects. Similar to dropout, it adds some noise to each hidden layer's activations. Therefore, if we use batch normalization, we will use less dropout, which is a good thing because we are not going to lose a lot of information. However, we should not depend only on batch normalization for regularization; we should better use it together with dropout [32].

3.6 Fully Connected Layer

In neural networks, each neuron receives input from some number of locations in the previous layer. In a fully connected layer, each neuron receives input from every element of the previous layer. In a convolutional layer, neurons receive input from only a restricted subarea of the previous layer. Typically the subarea is of a square shape (e.g., size 5 by 5). The input area of a neuron is called its receptive field. So, in a fully connected layer, the receptive field is the entire previous layer. In a convolutional layer, the receptive area is smaller than the entire previous layer [23]. Fully connected layer applies weights over the input generated by the feature analysis to predict an accurate label.

3.7 Softmax

The softmax function, also known as softargmax or normalized exponential function[33], is a generalization of the logistic function to multiple dimensions. It is used in multinomial logistic regression and is often used as the last activation function of a neural network to normalize the output of a network to a probability distribution over predicted output classes [34].

Softmax turn logits (numeric output of the last linear layer of a multi-class classification neural network) into probabilities by take the exponents of each output and then normalize each number by the sum of those exponents so the entire output vector adds up to one — all probabilities should add up to one. Cross entropy loss is usually the loss function for such a multi-class classification problem. That is, prior to applying softmax, some vector components could be negative, or greater than one; and might not sum to 1; but after applying softmax, each component will be in the interval { (0,1)}, and the components will add up to 1, so that they can be interpreted as probabilities[34].

Softmax is frequently appended to the last layer of an image classification network such as those in CNN (VGG16 for example) used in ImageNet competitions [35].

3.8 Adam optimization

The Adam optimization algorithm is an extension to stochastic gradient descent that can be used instead of the classical stochastic gradient descent procedure to update network weights iterative based in training data.

Adam combines the best properties of the Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp) algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems. So Instead of adapting the parameter learning rates based on the average first moment (the mean) as in RMSProp, Adam also makes use of the average of the second moments of the gradients (the uncentered variance). It is relatively easy to configure where the default configuration parameters do well on most problems [36].

3.9 Network Architectures

"Transfer Learning" which enables the use of pre-trained models from other people by making small changes to either improve the existing model or test another model against it by taking a pre-trained image classification network that has already learned to extract powerful and informative features from natural images and using it as a starting point to perform a new task. A pre-trained model may not be 100% accurate in some applications, but it saves huge efforts required to re-invent the wheel. There are different state-of-art networks that can be used, we choose VGG16, ResNet, MobileNet, Inception-v3, and Xception becouse of their performance, but also because they were examples that introduced specific architectural innovations.

4.1 Previous Studies

In the recent studies on the GI tract, most of the research is limited to a very specific abnormality such as polyps detection in the colon. This may be due to the shortage of medical data or ease of narrow down the scope or limit to more specialized solutions.

Authors in [53] proposed a system to detect polyps in colonoscopy. This system is able to generate an alert with near real-time feedback during colonoscopy. The authors use visual features and a rule-based classifier to detect the edges of polyps. This approach had achieved 97.7% accuracy in polyps detection.

The study in [54] described a system to classify abnormalities such as polyps, bleeding in colonoscopy videos. The authors had used a texton histogram of an image block in this approach. They had achieved 91% recall and 90.8% specificity on colonoscopy images.

The study of [55] proposed an approach to classify GI diseases and the authors used KVASIR dataset [5] for evaluating their approach. They use 6 pre-computed visual features per each image as provided in the dataset [56] and generate the texture of the images using local binary patterns [57] and Haralick features [58]. After the feature selection, the authors train a model using logistic regression and kernel discriminant analysis using spectral regression for each feature. Then they had applied ensemble technique on predictions. Through this approach, they had achieved accuracy of 94% with the F1-score of 0.76.

The study in [59] described a system to classify GI diseases on a similar dataset [56]. The authors of this research use inceptionlike CNN architecture and fixed-crop data augmentation for this classification. The proposed architecture is based on GoogLeNet. The authors had obtained an accuracy of 93.9% and F1-score of 0.755 for their best-fitted model.

The authors in [62] presented a survey of existing CADx systems that have been developed for detection of gastric abnormalities. The authors reviewed them based on their feature extraction techniques. These features extraction techniques were grouped based on their respective domain and descriptors. They also mentioned various endoscopy modalities, and abnormalities. The survey also discussed various open issues, trends, and challenges. Moreover, image data-sets associated with these medical-imaging challenges in the field of computer-assisted endoscopy were described.

Authors in the study [63] presented five different methods for the multi-class classification of GI tract diseases, two based on global feature extractions, and three based on CNN with transfer learning. The proposed approach based on the global features and pre-trained CNN with transfer learning mechanism. The authors found that the method of combination of two neural networks (Resnet-152 and Densenet-161) with an additional MLP achieved the highest performance with both the validation dataset and the test dataset provided by the task organizers with an accuracy rate of 95.80%, a precision of 95.87%, and an F1-score of 95.80%.

Authors in [64], developed an artificial intelligence (AI) system that automatically detects early signs of colorectal cancer during colonoscopy; the AI system showed the sensitivity and specificity were 97.3% and 99.0% respectively, and the area under the curve was 0.975 (95% CI=0.964–0.986) in the validation set. Moreover, the sensitivities were 98.0% (95% CI=96.6%–98.8%) in the polypoid subgroup and 93.7% (95% CI=87.6%–96.9%) in the nonpolypoid subgroup; To accelerate the detection, tensor metrics in the trained model was decomposed, and the system can predict cancerous regions 21.9ms/image on average.

The study of [65] used a deep-learning algorithm by using data from 1,290 patients, and validated it on newly collected 27,113 colonoscopy images from 1,138 patients with at least one detected polyp (per-image-sensitivity, 94.38%; per-image-specificity, 95.92%; area under the receiver operating characteristic curve, 0.984), on a public database of 612 polyp-containing images (per-image-sensitivity, 88.24%), on

138 colonoscopy videos with histologically confirmed polyps (per-image-sensitivity of 91.64%; per-polyp-sensitivity, 100%), and on 54 unaltered full-range colonoscopy videos without polyps (per-image-specificity, 95.40%). By using a multi-threaded processing system, the algorithm could process at least 25 frames per second with a latency of 76.80 ± 5.60ms in real-time video analysis.

The study in [60] proposed a framework for modular and automatic preprocessing of gastrointestinal tract images (MAPGI), which applies edge removal, contrast enhancement, filtering, color mapping, and scaling to each image in the Kvasir dataset[56]. Three state-of-the-art neural networks architectures, Inception-ResNet-v2, Inception-v4, and NASNet, are trained on the Kvasir dataset[56], and resulting accuracies achieved using Inception-v4, Inception-ResNet-v2, and NASNet were 0.9845, 0.9848, and 0.9735, respectively. In addition, Inception-v4 achieved an average of 0.938 precision, 0.939 recall, 0.991 specificity, 0.938 F1 score, and 0.929 Matthews correlation coefficient (MCC). Bootstrapping provided NASNet, the worst performing model, a lower bound of 0.9723 accuracies on the 95% confidence interval.

The authors in [66] proposed a combinations of three pre-trained CNNs namely DenseNet201, ResNet-18, and VGG-16 followed by a GAP layer are used for feature extraction. Furthermore, the authors described a dimensional reduction method to reduce processing time whereas preserving classification accuracy by consuming extracted feature vectors through truncated SVD. Then these feature vectors were used to train an ANN, SVM and Random Forest classifiers. They obtained the best classification accuracy for a single hidden layer ANN classifier. The proposed approach showed a promising accuracy of over 97%, which was a significant performance gain relating to the state-of-the-art approaches.

4.2 Comment about previous Studies

In summary, the first two literature is limited to a specific disease diagnosis namely polyp detection. The study of [55] and the study of [59] are based on pre-computed visual features and a deep CNN architecture respectively, which give lower F1-measures. Authors in [62] used feature extraction techniques for detection of gastric abnormalities. And in [63], multi-class classification of GI tract diseases was proposed via global feature extraction techniques and combination of three different pre-trained networks. The study of [64] and [65] showed good results in detecting early signs of colorectal cancer and polyps during colonoscopy respectively. The study of [60] trained state-of-the-art neural network architectures on the Kvasir dataset [56] and apply digital data processing on the images. And the study [66] attempted different approaches of feature extraction by using different combinations of pre-trained CNNs with three classifiers (ANN, SVM, and Random Forest). Moreover, we aim to increase the resulted accuracy in a comprehensive ensemble model for GI tract disease and anatomical landmarks classification using some of the current state-of-the-art approaches.

5.1 Dataset

The initial Kvasir dataset consists of 8,000 images, annotated and verified by medical doctors (experienced endoscopists), including 8 classes showing anatomical landmarks, phatological findings or endoscopic procedures in the GI tract. 1000 images for each class. The anatomical landmarks are Z-line, cecum, and pylorus, while the pathological finding includes esophagitis, polyps and ulcerative colitis. In addition, Kvasir dataset provide two set of images related to removal of polyps, the "dyed and lifted polyp" and the "dyed resection margins". Some of the included classes of images have a green picture in picture illustrating the position and configuration of the endoscope inside the bowel, by use of an electromagnetic imaging system (ScopeGuide, Olympus Europe) that may support the interpretation of the image [56]. Two different samples from two different categories can be seen in Fig. (4.1)(a) and (b).

(a) Abnormal Polyps

(b): Normal-pylorus

Fig. (4.1): Normal and abnormal pylorus [56]

5.2 Language and tool used

All the implementations that applied in this thesis used Python language as the main programing language. The

implementations has been coded and tested in colab by google that provide us a GPU for free.

5.3 Image format

The dataset consist of RGB images with different resolution from 720x576 up to 1920x1072 pixels. All the images have been reprocessed to be with same size of 256x256.

5.4 Preprocessing

Data should be formatted into appropriately pre-processed floating point tensors before being fed into our network by using normalization. Currently, our data sits on a drive as JPEG files, so the steps for getting it into our network are roughly:

- Read the picture files.
- Decode the JPEG content to RBG grids of pixels.
- Convert these into floating point tensors.
- Rescale the pixel values (between 0 and 255) to the [0, 1] interval (as you know, neural networks prefer to deal with small input values).

Keras has a module with image processing helper tools, located at keras.preprocessing.image. In particular, it contains the class ImageDataGenerator which allows to quickly set up Python generators that can automatically turn image files on disk into batches of pre-processed tensors. This is what we have used and applying laplacian filters.

5.5 Data augmentation

Deep networks need a large amount of training data to achieve good performance. Image augmentation can increase the size of the training set without acquiring new images and build a powerful image classifier using very little training data, image augmentation is usually required to boost the performance of deep networks. Image augmentation artificially creates training images through different ways of processing or a combination of multiple processing, such as random rotation, shifts, shear, and flips, random cropping, etc.

The Keras deep learning neural network library provides the capability to fit models using image data augmentation via the ImageDataGenerator class that generates batches of image data with real-time data augmentation.

In order to make the most of our few training examples, increase the accuracy of the model, and avoid overfitting, we augmented the data via a number of random transformations. The selected data augmentation techniques are listed below:

```
rotation_range=20,
width_shift_range=0.2,
height_shift_range=0.2,
horizontal_flip=True,
vertical_flip=True
```

5.6 Network Architecture

A pre-trained network is simply a saved network previously trained on a large dataset, typically on a large-scale image classification task. If this original dataset is large enough and general enough, then the spatial feature hierarchy learned by the pre-trained network can effectively act as a generic model of our visual world, and hence its features can prove useful for many different computer vision problems, even though these new problems might involve completely different classes from those of the original task. For instance, one might train a network on and then re-purpose this trained network for something as remote as identifying furniture items in images. Such portability of learned features across different problems is a key advantage of deep learning compared to many older shallow learning approaches, and it makes deep learning very effective for small-data problems.

There are two ways to leverage a pre-trained network: *feature extraction* and *fine-tuning*. Feature extraction consists of using the representations learned by a previous network to extract interesting features from new samples. These features are then run through a new classifier, which is trained from scratch. Fine-tuning consists in unfreezing a few of the top layers of a frozen model base used for feature extraction, and jointly training both the newly added part of the model and these top layers. This is called "fine-tuning" because it slightly adjusts the more abstract representations of the model being reused, in order to make them more relevant for the problem at hand.

Convnets used for image classification comprise two parts: they start with a series of pooling and convolution layers, and they end with a densely-connected classifier. The first part is called the "convolutional base" of the model. In the case of convnets, extracting features will simply consist of taking the convolutional base of a previously-trained network and extending it by adding new classifier (Dense layers) on top, and running the whole thing end-t-end on the input data as shown in fig. (4.2). This allows using data augmentation, because every input image is going through the convolutional base every time it is seen by the model. However, for this same reason, this technique is in fact so expensive that should only attempt by having access to a GPU: it is absolutely intractable on CPU.

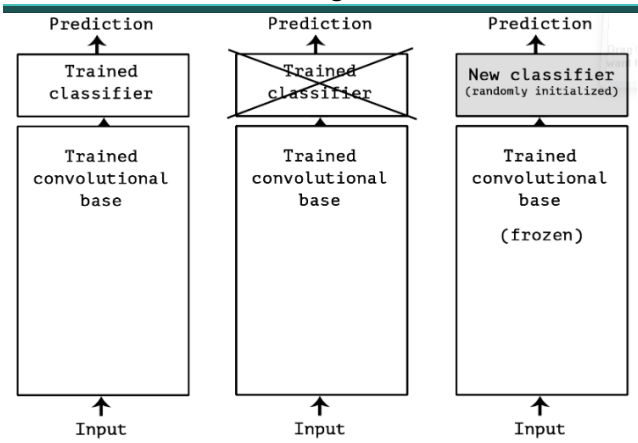


Fig. (4.2): taking the conv_base of a previously-trained network and extending it by adding new classifier [22]

We have experimented with a variety of pre-trained models that comes pre-packaged with Keras application library and available as part of the keras.application module, namely VGGNets[37], ResNet[40], MobileNet[43], Inception[44], and Xception[46]. We leverage each one of the five pre-trained network with feature extraction and fine tuning technique by extending each of the five model to have (conv_base) by adding Dense layers on top, and running the whole thing end-to-end on the input data. These techniques allow us to use data augmentation for better training experiment and avoiding overfitting that might face us while testing our model. In compilation our networks we choose Adam optimizer algorithm with learning rate of 0.0001. Softmax function with cross entropy loss function have been chosen for our multi-class classification problem. For evaluation, the Hold-out technique was applied and F-score metric has been chosen to be our evaluation metric. We have done our implementation on google colaboratory to take the advantage of using free GPU.

5.6.1 VGG16

The architecture of the VGG16 convolutional base consist of six blocks, each block consist of a sequence of conv2D and maxPooling layers. We add BatchNormalization layer, Relu activation function, Flatten layer, and Dense layer with Softmax activation function. We made data augmentation as mentioned in section 4.5 on our model to avoid overfitting and get better accuracy.

5.6.1.1 Training the VGG16 Model

After training the model with 20 epochs and 128 as a batch size, we achieved an accuracy of 90.06% and loss of 0.2835 on Kvasir training set. Progress for training loss and accuracy can be shown in fig. (4.3) below:

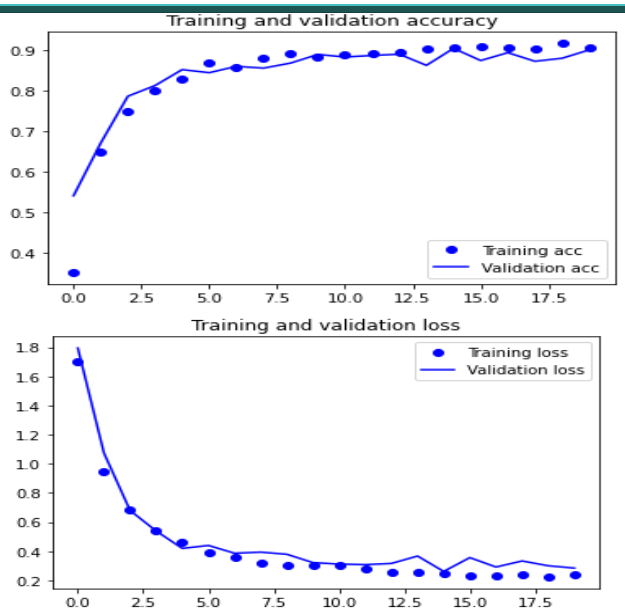


Fig. (4.3): Progress for VGG16 training loss and accuracy

5.6.1.2 Validation of the VGG16 Model

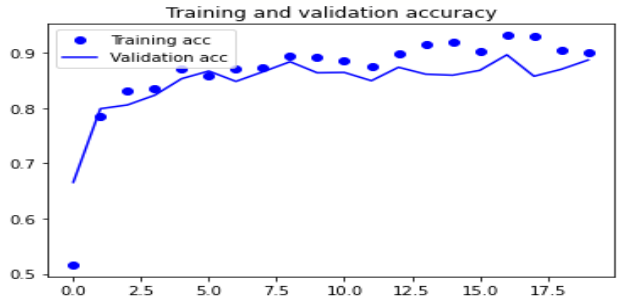
After testing the Model on the Kvasir tset set, our model found 7866 of 8000 images belonging to 8 classes, and achieve accuracy of 98.32%.

5.6.2 ResNet

We add GlobalMaxPooling2D layer and Dense layer with Softmax activation function on top of ResNet (conv_base), and running the whole thing end-to-end on the input data. Also we apply different augmentation techniques on our model to make the most of our few training example, avoid overfitting and get better accuracy.

5.6.2.1 Training the ResNet Model

After training the model with 20 epochs and 64 as a batch size, we achieved an accuracy of 88.69% loss of 0.6235 on Kaggle training set. Progress for training loss and accuracy can be shown in fig. (4.4) below:



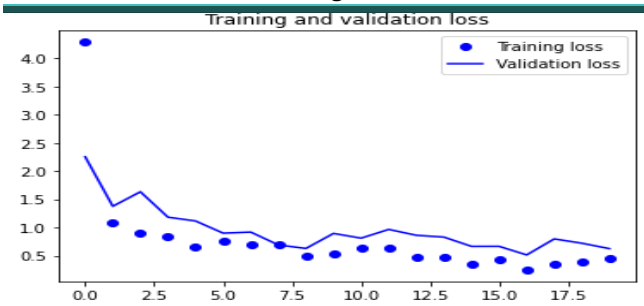


Fig. (4.4): Progress for ResNet training loss and accuracy

5.6.2.2 Validation of the ResNet Model

After testing the Model on the Kvasir tset set, our model found 7385 of 8000 images belonging to 8 classes, and achieve accuracy of 92.3125% .

5.6.3 MobileNet

We add GlobalMaxPooling2D layer and Dense layer with Softmax activation function on top of MobileNet (conv_base), and running the whole thing end-to-end on the input data. We made data augmentation with different augmentation techniques on our model to make the most of our few training example, avoid overfitting and get better accuracy.

5.6.3.1 Training the MobileNet Model

After training the model with 20 epochs and 64 as a batch size, we achieved an accuracy of 85.12% and loss of 0.5149 on Kvasir training set. Progress for training accuracy and loss can be shown in fig. (4.5) below:

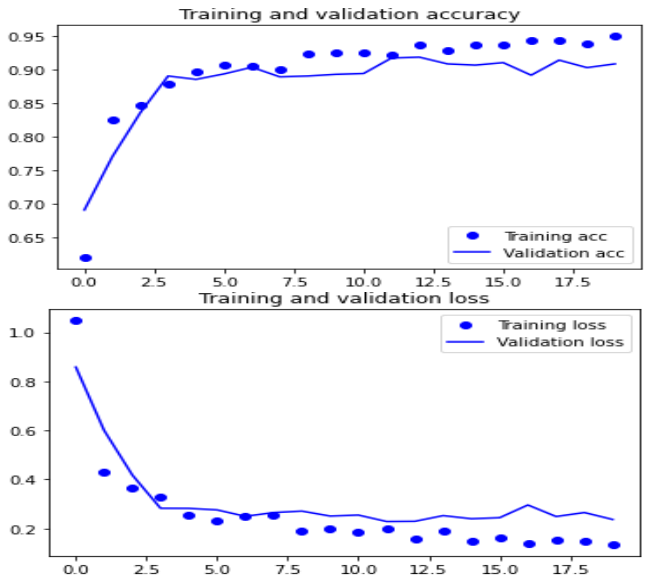


Fig. (4.5): Progress for MobileNet training loss and accuracy

5.6.3.2 Validation of the MobileNet Model

After testing the Model on the Kvasir tset set, our model found 7811 of 8000 images belonging to 8 classes, and achieve accuracy of 97.67% .

5.6.4 InceptionV3

We add GlobalMaxPooling2D layer and Dense layer with Softmax activation function on top of InceptionV3 (conv_base), and running the whole thing end-to-end on the input data. We made data augmentation on our model to prevent overfitting and get better accuracy.

5.6.4.1 Training the InceptionV3 Model

After training the model with 20 epochs and 64 as a batch size, we achieved an accuracy of 89.31% and loss of 0.3168 on Kvasir training set. Progress for training accuracy and loss can be shown in fig. (4.6) below:

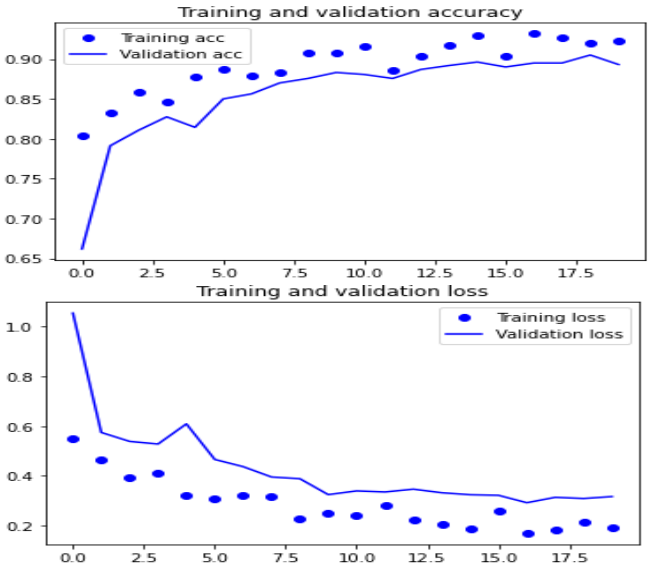


Fig. (4.6): Progress for InceptionV3 training loss and accuracy

5.6.4.2 Validation of the InceptionV3 Model

After testing the Model on the Kvasir test set, our model found 7200 of 8000 images belonging to 8 classes, and achieve accuracy of 90.00% .

4.6.5 Xception

We add GlobalMaxPooling2D layer and Dense layer with Softmax activation function on top of Xception (conv_base), and running the whole thing end-to-end on the input data. We made data augmentation on our model to prevent overfitting and get better accuracy.

5.6.5.1 Training the Xception Model

After training the model with 20 epochs and 32 as a batch size, we achieved an accuracy of 90.87% and loss of

0.2367 on Kvasir training set. Progress for training accuracy and loss can be shown in fig. (4.7) below :

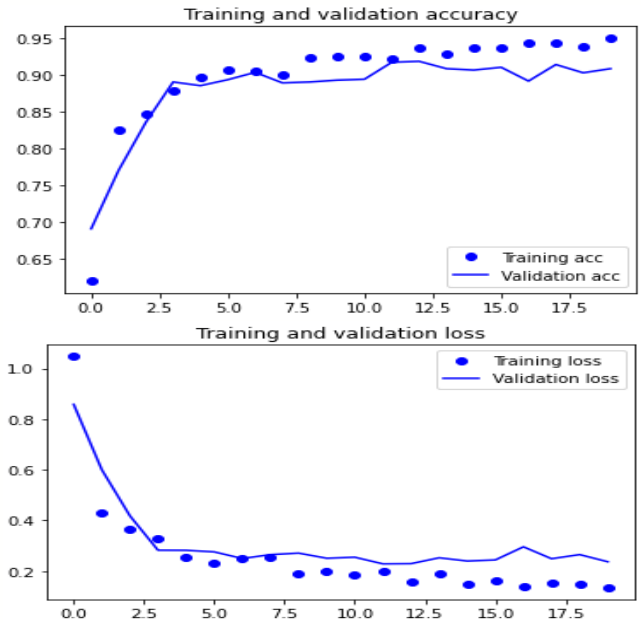


Fig. (4.7): Progress for Xception training loss and accuracy

5.6.5.2 Validation of the Xception Model

After testing the Model on the Kvasir test set, our model found 7862 of 8000 images belonging to 8 classes, and achieve accuracy of 98.27% .

6. Evaluation of the Model

6.1 Data Set for testing of each model

Each model has been testing on a 25% from the Kvasir dataset as a test set. And each model test all over the eight categories of the dataset with 1000 image to each category.

6.2 Testing each model

Since we want our submissions to be as close as possible to the actual probabilities and we have a number of classes, we used the categorical cross-entropy loss function with softmax as the last layer's activation function. Since the dataset is relatively small, we had to use transfer learning with a set of pre-trained models, focusing on VGGNets[37], ResNet[40], MobileNet[43], Inception[44], and Xception[46] of various sizes that were pre-trained on ImageNet[38].

In order to apply these pre-trained models to a new dataset, we replace the last fully-connected layer; this is necessary to output the classes of the new inputs. In our case, we replaced the last fully-connected layer with GlobalMaxPooling2D layer, and fully-connected dense layer with a softmax activation function, and in the case of VGG16 network we add BatchNormalization layer, Relu activation function, Flatten layer, and Dense layer with Softmax activation function. After that, we re-train all the pre-trained layers at maximum of 20 epochs. Each model from the five models has been tested on a test dataset with 8 classes, each class with 1000 image.

In order to make the most of our few training examples and increase the accuracy of the model, we augmented the data via a number of random transformations. Furthermore, it is expected that data augmentation should also help prevent overfitting which is a common problem with small datasets, when the model, exposed to too few examples, learns patterns that do not generalize to new data, and for this reason, improving the models ability to generalize.

6.3 Result and Discussion

Each Model has achieved different accuracy than the other when testing on the test dataset provided by Kvasir, but the most accurate result has been achieved when we retrain VGG16 and Xception with accuracy reach to 98.32499%. The structure of these models and since they are achieve high performance when trained on ImageNet data make their results more accurate and score a high f-measure values as a validation metric in our classification study. The resulted accuracy of each model is shown in the table 1 below:

Table 1: Accuracy of models

Model Name	Test accuracy
VGG16	98.3249%8
ResNet	92.3125%
MobileNet	97.63%
InceptionV3	90.0%
Xception	98.275%

7.1 Conclusion

This thesis presents methodologies for maximizing classification accuracy on a small medical image dataset, the Kvasir dataset, by performing robust image preprocessing and applying five state-of-the-art deep learning which are VGG, ResNet, MobileNet, Inception-v3, and Xception. Images are classified as being or involving an anatomical landmark (pylorus, z-line, cecum), a diseased state (esophagitis, ulcerative colitis, polyps), or a medical procedure (dyed lifted polyps, dyed resection margins). The resulted accuracy after testing each model shows that VGG model was the best model with accuracy of 98.3%.

7.2 Future Work

Future work can involve enhancing the performance of the models by using some of the digital image processing on the dataset, and make some model evaluation processes to estimate performance. Other state-of-art deep learning can be involved in training the kvasir dataset.

References

1. Al Barsh, Y. I., et al. (2020). "MPG Prediction Using Artificial Neural Network." International Journal of Academic Information Systems Research (IJAISR) 4(11): 7-16.
2. Alajrami, E., et al. (2019). "Blood Donation Prediction using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 3(10): 1-7.
3. Alajrami, E., et al. (2020). "Handwritten Signature Verification using Deep Learning." International Journal of Academic Multidisciplinary Research (IJAMR) 3(12): 39-44.
4. Al-Araj, R. S. A., et al. (2020). "Classification of Animal Species Using Neural Network." International Journal of Academic Engineering Research (IJAER) 4(10): 23-31.
5. Al-Atrash, Y. E., et al. (2020). "Modeling Cognitive Development of the Balance Scale Task Using ANN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 74-81.
6. Alghoul, A., et al. (2018). "Email Classification Using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 2(11): 8-14.
7. Abu Nada, A. M., et al. (2020). "Age and Gender Prediction and Validation through Single User Images Using CNN." International Journal of Academic Engineering Research (IJAER) 4(8): 21-24.
8. Abu Nada, A. M., et al. (2020). "Arabic Text Summarization Using AraBERT Model Using Extractive Text Summarization Approach." International Journal of Academic Information Systems Research (IJAISR) 4(8): 6-9.
9. Abu-Saqer, M. M., et al. (2020). "Type of Grapefruit Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 4(1): 1-5.
10. Afana, M., et al. (2018). "Artificial Neural Network for Forecasting Car Mileage per Gallon in the City." International Journal of Advanced Science and Technology 124: 51-59.
11. Al-Araj, R. S. A., et al. (2020). "Classification of Animal Species Using Neural Network." International Journal of Academic Engineering Research (IJAER) 4(10): 23-31.
12. Al-Atrash, Y. E., et al. (2020). "Modeling Cognitive Development of the Balance Scale Task Using ANN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 74-81.
13. Alghoul, A., et al. (2018). "Email Classification Using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 2(11): 8-14.
14. Al-Kahlout, M. M., et al. (2020). "Neural Network Approach to Predict Forest Fires using Meteorological Data." International Journal of Academic Engineering Research (IJAER) 4(9): 68-72.
15. Alkronz, E. S., et al. (2019). "Prediction of Whether Mushroom is Edible or Poisonous Using Back-propagation Neural Network." International Journal of Academic and Applied Research (IJAAR) 3(2): 1-8.
16. Al-Madhoun, O. S. E.-D., et al. (2020). "Low Birth Weight Prediction Using JNN." International Journal of Academic Health and Medical Research (IJAHMR) 4(11): 8-14.
17. Al-Massri, R., et al. (2018). "Classification Prediction of SBRCTs Cancers Using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 2(11): 1-7.
18. Al-Mobayed, A. A., et al. (2020). "Artificial Neural Network for Predicting Car Performance Using JNN." International Journal of Engineering and Information Systems (IJEAIS) 4(9): 139-145.
19. Al-Mubayyed, O. M., et al. (2019). "Predicting Overall Car Performance Using Artificial Neural Network." International Journal of Academic and Applied Research (IJAAR) 3(1): 1-5.
20. Alshawwa, I. A., et al. (2020). "Analyzing Types of Cherry Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 4(1): 1-5.
21. Al-Shawwa, M., et al. (2018). "Predicting Temperature and Humidity in the Surrounding Environment Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(9): 1-6.
22. Ashqar, B. A., et al. (2019). "Plant Seedlings Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 3(1): 7-14.
23. Bakr, M. A. H. A., et al. (2020). "Breast Cancer Prediction using JNN." International Journal of Academic Information Systems Research (IJAISR) 4(10): 1-8.
24. Barhoom, A. M., et al. (2019). "Predicting Titanic Survivors using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 3(9): 8-12.
25. Belbeisi, H. Z., et al. (2020). "Effect of Oxygen Consumption of Thylakoid Membranes (Chloroplasts) From Spinach after Inhibition Using JNN." International Journal of Academic Health and Medical Research (IJAHMR) 4(11): 1-7.
26. Dalffa, M. A., et al. (2019). "Tic-Tac-Toe Learning Using Artificial Neural Networks." International Journal of Engineering and Information Systems (IJEAIS) 3(2): 9-19.
27. Dawood, K. J., et al. (2020). "Artificial Neural Network for Mushroom Prediction." International Journal of Academic Information Systems Research (IJAISR) 4(10): 9-17.
28. Dheir, I. M., et al. (2020). "Classifying Nuts Types Using Convolutional Neural Network." International Journal of Academic Information Systems Research (IJAISR) 3(12): 12-18.
29. El-Khatib, M. J., et al. (2019). "Glass Classification Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 3(2): 25-31.
30. El-Mahelawi, J. K., et al. (2020). "Tumor Classification Using Artificial Neural Networks." International Journal of Academic Engineering Research (IJAER) 4(11): 8-15.
31. El-Mashharawi, H. Q., et al. (2020). "Grape Type Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 41-45.
32. Elzamly, A., et al. (2015). "Classification of Software Risks with Discriminant Analysis Techniques in Software planning Development Process." International Journal of Advanced Science and Technology 81: 35-48.
33. Elzamly, A., et al. (2015). "Predicting Software Analysis Process Risks Using Linear Stepwise Discriminant Analysis: Statistical Methods." Int. J. Adv. Inf. Sci. Technol 38(38): 108-115.
34. Elzamly, A., et al. (2017). "Predicting Critical Cloud Computing Security Issues using Artificial Neural Network (ANNs) Algorithms in Banking Organizations." International Journal of Information Technology and Electrical Engineering 6(2): 40-45.
35. Habib, N. S., et al. (2020). "Presence of Amphibian Species Prediction Using Features Obtained from GIS and Satellite Images." International Journal of Academic and Applied Research (IJAAR) 4(11): 13-22.
36. Harz, H. H., et al. (2020). "Artificial Neural Network for Predicting Diabetes Using JNN." International Journal of Academic Engineering Research (IJAER) 4(10): 14-22.
37. Hassanein, R. A. A., et al. (2020). "Artificial Neural Network for Predicting Workplace Absenteeism." International Journal of Academic Engineering Research (IJAER) 4(9): 62-67.
38. Heriz, H. H., et al. (2018). "English Alphabet Prediction Using Artificial Neural Networks." International Journal of Academic Pedagogical Research (IJAPR) 2(11): 8-14.
39. Jaber, A. S., et al. (2020). "Evolving Efficient Classification Patterns in Lymphography Using EasyNN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 66-73.
40. Kashf, D. W. A., et al. (2018). "Predicting DNA Lung Cancer using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(10): 6-13.
41. Khalil, A. J., et al. (2019). "Energy Efficiency Predicting using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 3(9): 1-8.
42. Kweik, O. M. A., et al. (2020). "Artificial Neural Network for Lung Cancer Detection." International Journal of Academic Engineering Research (IJAER) 4(11): 1-7.
43. Maghari, A. M., et al. (2020). "Books' Rating Prediction Using Just Neural Network." International Journal of Engineering and Information Systems (IJEAIS) 4(10): 17-22.
44. Mettleq, A. S. A., et al. (2020). "Mango Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 22-29.
45. Metwally, N. F., et al. (2018). "Diagnosis of Hepatitis Virus Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(11): 1-7.
46. Mohammed, G. R., et al. (2020). "Predicting the Age of Abalone from Physical Measurements Using Artificial Neural Network." International Journal of Academic and Applied Research (IJAAR) 4(11): 7-12.
47. Musleh, M. M., et al. (2019). "Predicting Liver Patients using Artificial Neural Network." International Journal of Academic Information Systems Research (IJAISR) 3(10): 1-11.
48. Oriban, A. J. A., et al. (2020). "Antibiotic Susceptibility Prediction Using JNN." International Journal of Academic Information Systems Research (IJAISR) 4(11): 1-6.
49. Qwaider, S. R., et al. (2020). "Artificial Neural Network Prediction of the Academic Warning of Students in the Faculty of Engineering and Information Technology in Al-Azhar University-Gaza." International Journal of Academic Information Systems Research (IJAISR) 4(8): 16-22.
50. Sadek, R. M., et al. (2019). "Parkinson's Disease Prediction Using Artificial Neural Network." International Journal of Academic Health and Medical Research (IJAHMR) 3(1): 1-8.
51. Salah, M., et al. (2018). "Predicting Medical Expenses Using Artificial Neural Network." International Journal of Engineering and Information Systems (IJEAIS) 2(20): 11-17.
52. Salman, F. M., et al. (2020). "COVID-19 Detection using Artificial Intelligence." International Journal of Academic Engineering Research (IJAER) 4(3): 18-25.
53. Samra, M. N. A., et al. (2020). "ANN Model for Predicting Protein Localization Sites in Cells." International Journal of Academic and Applied Research (IJAAR) 4(9): 43-50.
54. Shawarib, M. Z. A., et al. (2020). "Breast Cancer Diagnosis and Survival Prediction Using JNN." International Journal of Engineering and Information Systems (IJEAIS) 4(10): 23-30.
55. Zaqout, I., et al. (2015). "Predicting Student Performance Using Artificial Neural Network: in the Faculty of Engineering and Information Technology." International Journal of Hybrid Information Technology 8(2): 221-228.
56. Saleh, A. et al. (2020). "Brain Tumor Classification Using Deep Learning." 2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech). IEEE, 2020.
57. Almadhoun, H. et al. (2021). "Classification of Alzheimer's Disease Using Traditional Classifiers with Pre-Trained CNN." International Journal of Academic Health and Medical Research (IJAHMR) 5(4):17-21.
58. El-Habil, B. et al. (2022). "Cantaloupe Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 5(12): 7-17.
59. Alkahlout, M. A. et al. (2022). "Classification of Fruits Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 5(12): 56-63.
60. Alfarra, A. H. et al. (2022). "Classification of Pineapple Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 5(12): 37-41.
61. Al-Masawabe, M. M. et al. (2022). "Papaya maturity Classification Using Deep Convolutional Neural Networks." International Journal of Engineering and Information Systems (IJEAIS) 5(12): 60-67.
62. Abu-Jamie, T. N., et al. (2022). "Six Fruits Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 6(1): 1-8.
63. Aish, M. A., et al. (2022). "Classification of pepper Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(1): 24-31.
64. Alsaqqa, A. H., et al. (2022). "Using Deep Learning to Classify Different types of Vitamin." International Journal of Academic Engineering Research (IJAER) 6(1): 1-6.
65. M. Alnajjar, M. K. and S. S. Abu-Naser (2022). "Heart Sounds Analysis and Classification for Cardiovascular Diseases Diagnosis using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(1): 7-23.
66. Aldammagh, Z., Abdeljawad, R., & Obaid, T. (2021). Predicting Mobile Banking Adoption: An Integration of TAM and TPB with Trust and Perceived Risk. Financial Internet Quarterly, 17(3), 35-46.
67. Obaid, T. (2021). Predicting Mobile Banking Adoption: An Integration of TAM and TPB with Trust and Perceived Risk. Available at SSRN 3761669.
68. Jouda, H., Abu Jarad, A., Obaid, T., Abu Mdallalah, S., & Awaja, A. (2020). Mobile Banking Adoption: Decomposed Theory of Planned Behavior with Perceived Trust. Available at SSRN 3660403.
69. Obaid, T., Abdaljawad, R., & Mdallalahc, S. A. (2020). Factors Driving E-Learning Adoption In Palestine: An Integration of Technology Acceptance Model And IS Success Model. Available at SSRN 3686490.
70. Obaid, T. F., & Eneizan, B. M. (2016). Transfer of training and post-training on job performance in Middle Eastern countries. Review of Public Administration and Management, 400(3786), 1-11.
71. Obaid, T. F., Zainon, M. S., Eneizan, P. D. B. M., & Wahab, K. A. TRANSFER OF TRAINING AND POST-TRAINING ON JOB PERFORMANCE IN MIDDLE EASTERN COUNTRIES.
72. Musleh, M. M., et al. (2019). "Predicting Liver Patients using Artificial Neural Network." International Journal of Academic Information Systems Research (IJAISR) 3(10): 1-11.
73. Qwaider, S. R., et al. (2020). "Artificial Neural Network Prediction of the Academic Warning of Students in the Faculty of Engineering and Information Technology in Al-Azhar University-Gaza." International Journal of Academic Information Systems Research (IJAISR) 4(8): 16-22.
74. Mettleq, A. S. A., et al. (2020). "Mango Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 22-29.
75. Salman, F. M., et al. (2020). "COVID-19 Detection using Artificial Intelligence." International Journal of Academic Engineering Research (IJAER) 4(3): 18-25.
76. Metwally, N. F., et al. (2018). "Diagnosis of Hepatitis Virus Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(11): 1-7.

77. R. Nawarathna, J. Oh, J. Muthukudage, W. Tavanapong, J. Wong, P. C. De Groen, and S. J. Tang, "Abnormal image detection in endoscopy videos using a filter bank and local binary patterns," Neurocomputing, vol. 144, pp. 70–91, 2014.
78. The kvasir dataset. [Online]. Available: https://datasets.simula.no/kvasir/, Accessed: 2020-10-25
79. T. Ojala, M. Pietikainen, and T. M ¨ aenp ¨ a¨a, "Multiresolution gray-scale ¨ and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis & Machine Intelligence, no. 7, pp. 971–987, 2002.
80. R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," IEEE Transactions on systems, man, and cybernetics, no. 6, pp. 610–621, 1973.
81. Cogan, Timothy & Cogan, Maribeth & Tamil, Lakshman. (2019). MAPGITI: Accurate identification of anatomical landmarks and diseased tissue in gastrointestinal tract using deep learning. Computers in Biology and Medicine. 111. 103351. 10.1016/j.compbiomed.2019.103351.
82. Chollet, Francois. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. 1800-1807. 10.1109/CVPR.2017.195.
83. Thambawita, Vajira & Jha, Debesh & Riegler, Michael & Halvorsen, Pål & Hammer, Hugo & Johansen, Håvard & Johansen, Dag. (2018). The Medico-Task 2018: Disease Detection in the Gastrointestinal Tract using Global Features and Deep Learning.
84. Wang, P., Xiao, X., Glissen Brown, J.R. et al. Development and validation of a deep-learning algorithm for the detection of polyps during colonoscopy. Nat Biomed Eng 2, 741–748 (2018). https://doi.org/10.1038/s41551-018-0301-3.
85. Gamage, Chathurika, et al. "Gi-net: Anomalies classification in gastrointestinal tract through endoscopic imagery with deep learning." 2019 Moratuwa Engineering Research Conference (MERCon). IEEE, 2019.