NTNU
Kunnskap for en bedre verden

Department of Computer Science

TDT4173 - Assignment 3

Image classification of abnormalities in the gastrointestinal tract: a comparison between CNN and KNN

Group:
Group 10 - Supervised Learning

Authors:
Kari L. Ness (kariln)
Tomas Samset (tomassam)
Sebastian M. Andresen (sebasman)

November 30, 2020

Abstract

This paper presents results from image classification by machine learning of medical images in the gastrointestinal tract. The aim is to locate and identify abnormalities automatically, which is a challenge in computer-aided diagnosis.

Automatic diagnostic systems are developed as aids for medical professionals as they are making a clinical diagnosis. The task of clinical diagnostics usually involves visual inspection or localization of abnormalities. Machine learning algorithms have proved to have good performance in image classification. They have successfully been implemented to identify abnormalities, thereby serving as a second opinion for medical professionals.

Two machine learning algorithms, convolutional neural networks (CNN) and k-nearest neighbor (KNN), have been used to identify disease and abnormalities on endoscopic images in the GI tract. CNN is a deep learning method and is regarded as the state-of-the-art method in computer-aided diagnostics. KNN is a simple, traditional machine learning algorithm and is explainable and easy to implement. CNN proved to perform well with an accuracy of 92%, which is close to comparable to how well the medical experts in the field perform. CNN clearly outperformed KNN in this classification task.

A code implementation in Python for both machine learning approaches in the project is publicly available at `https://github.com/tomassams/tdt4173-machine-learning-project`.

A website demonstrating activation maps of the CNN model is available at `https://tdt4173-cnn-visualization.netlify.app`.

Table of Contents

List of Figures

List of Tables

Nomenclature

$d_{minkowski}$	Minkowski distance
2-D	Two dimensional
AI	Artificial intelligence
ANN	Artificial neural networks
BGR	Blue, green, red
CAD	Computer-Aided Diagnostics
CNN	Convolutional Neural Network
CRN	Cancer Registry of Norway
d	Distance metric
DL	Deep learning
EDG	Endoscopy of the upper GI tract
GI	Gastrointestinal
HPO	Hyperparameter optimazation
K	Number of relevant neighbors in KNN
KNN	K-nearest neighbor
LSTM	Long Short-Term Memory
ML	Machine learning
NN	Neural network
p	hyperparameter in Minkowski distance.
ReLU	Rectified linear unit
ResNet	Residual network
ResNet50	Residual network with 50 layers
VV	Vestre Viken Hospital Trust

1 Introduction

Machine learning (ML) is a subfield of artificial intelligence (AI) and enables a computer to improve its performance through experience [36]. A machine learning challenge that has been widely explored and implemented is image recognition. Image recognition is a collective term used for computer technologies that can recognize objects or targeted subjects in images or videos. This recognition is done easily by humans, but has proved to be a task with significant complexity for computers [19].

One of the fields that have implemented image recognition through ML is the medical field, where computer-aided diagnostics (CAD) is one of today's buzzwords. CAD can serve as expert systems that utilize computer science to imitate diagnostic decisions made by skilled humans through diagnostic rules [61]. When determining a clinical diagnosis, medical professionals have to identify anatomical abnormalities in a patient. Identifying abnormalities usually involves visual inspection or manual localization of anatomical landmarks, requires much human effort, and is time-consuming. By implementing a ML algorithm trained to identify these abnormalities, computer-aided diagnostics can provide a second opinion for the medical professionals, and thereby reducing the risk of human error [51]. In addition to serving as an expert system, more complex systems have been developed to infer knowledge from analyzed data. It has proved useful to implement AI, machine learning, and data mining in automatic diagnostics systems when analyzing clinical data that is complex or massive [61].

About 2.8 million gastrointestinal (GI) cancer cases (esophageal, stomach, colorectal) are discovered globally every year [12]. With a mortality rate of about 50%, these diseases pose a significant health challenge, both on an individual and societal level [8]. The mortality rate related to the diseases is proved to be related to early detection, with a higher than 90% 5-year relative survival rate of cases detected in stage 1, and only 10% survival rate of disease discovered in stage 4 [8]. Because of this, efficient and precise diagnostics are vital for patient welfare. Endoscopy is the typical way to diagnose and treat diseases in the GI tract and involves direct inspection, biopsy, photography, and video recording [14]. Despite being an efficient method, endoscopic examinations are limited due to the human variation of the operator. This variation is related to operator skill, perceptual factors, personality characteristics, knowledge, and attitude, which may lead to failed diagnostics [23]. The operator variation can be reduced through educational efforts but will never be non-existing, and reducing failed diagnostics by minimizing operator variation with ML has high potential.

This project explores the supervised image classification of medical images from endoscopy of the GI tract with machine learning applications. For this task, two ML approaches were implemented; convolutional neural networks (CNN) and k-nearest neighbors (KNN). CNN is a complex state-of-the-art method, while KNN is a simple, traditional ML algorithm. In addition, the CNN is implemented both with and without residual networks. The classification is performed on Simula's Kvasir dataset, which contains labeled images of anatomical abnormalities inside the GI tract [44]. As there are no previous publications on implementations of KNN and CNN with residual networks on the Kvasir dataset, the project is novel work implemented with existing methods.

2 Related Work

The Kvasir dataset contains medical images from the GI tract. Medical imaging includes those processes that provide visual information of the human body. Today, all modern healthcare facilities utilize medical images to help medical professionals make the diagnostic and treatment process more efficient [2]. Abnormalities and diseases are identified through visual examination or localization of anatomical landmarks. Successful implementation of machine learning and image processing has been performed on several types of diseases, such as breast cancer [5, 38]. Even though diseases in the GI tract pose a significant risk to life expectancy and automatic diagnosis of diseases in the GI tract is a hot topic, today's automated diagnosis systems are still not competitive with an expert endoscopist [13, 59]. Several papers have been published, and a significant effort to successfully implement machine learning for automatic diagnosis in the GI tract has been made. However, many are not convincing due to small and non-public datasets, resulting in poor performance and reproducibility [57]. Several machine learning methods and algorithms have been tested in recent years, ranging from traditional ML methods such as KNN to more recently developed deep learning (DL) approaches.

Chu et al. [10] investigated eight classical, predictive ML models trained to identify and predict the necessary treatment of acute GI bleeding. All models were trained on clinical patient data, and the performance of the models was compared. They provided patient-specific recommendations with accuracies exceeding 70—80%, where Random Forest was the most accurate algorithm. KNN was significantly less accurate than the other methods.

The current state-of-the-art and the most commonly used image classification methods on medical images are based on deep learning networks, which are suitable for big data [13]. Papers recently published on the application of machine learning in GI diagnostics are usually based on deep learning, where CNNs are one of the most popular methods due to their high performance in image, video, and audio classification [34]. Takiyama et al. [56] showed promising results in the automatic anatomical classification of endoscopy images of the upper GI (EDG) using convolutional neural networks. The study stated that the trained CNN could locate specific anatomical locations within the stomach with an accuracy of 97.4%. In addition, several studies have had promising results in the localization of polyps in the GI tract. In 2020, Song et al. [53] presented a CNN-based CAD system with a polyp localization accuracy of 81.3–82.4%. Experts localize polyps with an accuracy of 82.4–87.3%, and the CAD results are almost comparable with the experts [53].

Most studies performed on automatic diagnostics in the GI tract mainly focus on detecting specific diseases or abnormalities, such as GI bleeding, EDG, and polyps. Even though these studies' results prove accurate, patients often suffer from more than one disease or abnormality, indicating that a CAD should be capable of accurate multi-class classification. Thambawita et al. [57] published an extensive study on ML applied to GI tract abnormality classification. The study implements several traditional machine learning and deep learning methods of several datasets. The deep learning methods significantly outperform traditional methods.

A big challenge in CAD of the GI tract is the limited number of labeled data. Because of this, the publication of the Kvasir dataset by Pogorelov et al. [44] in 2017 was important for the research involving CADs of the GI tract. Several studies on the Kvasir dataset have recently been published, and all the studies have been based on CNN implementations [1, 2, 7, 11, 16, 18, 31, 33, 43, 65]. The published studies mainly focus on different classical CNN variations by adding layers and updating or optimizing layer parameters. The latest paper based on the Kvasir dataset was published by Öztürk and Özkaya in October 2020. Öztürk and Özkaya built a CNN-based model combined with a Long Short-Term Memory (LSTM) structure. The model outperformed the state of the art methods with an accuracy of 97.90%, even when tested on datasets with a small number of labeled data and an imbalanced sample number between classes.

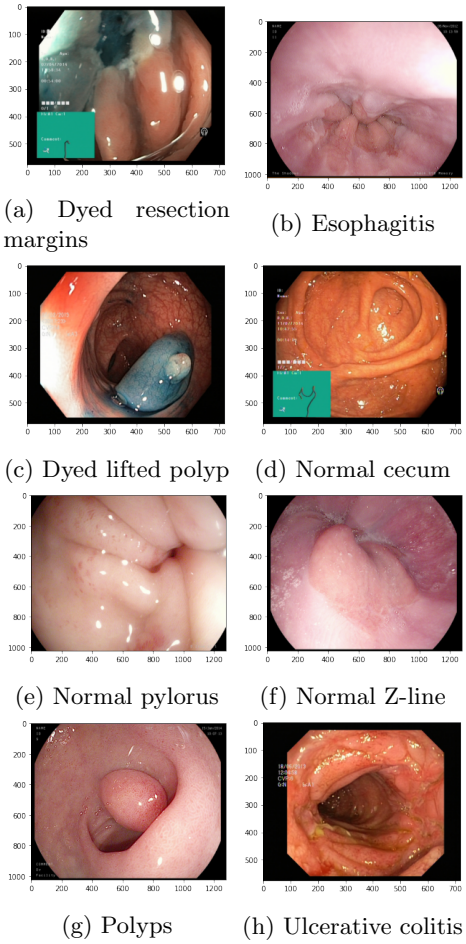
3 Data

The dataset analyzed in this project is the Kvasir dataset (version 2). The dataset was published in 2017 by Pogorelov et al. [44] and is a multi-class image dataset for computer-aided gastrointestinal disease detection. The dataset consists of medical images of the GI tract, which have been collected by Vestre Viken Hospital Trust (VV) in Norway through endoscopic examinations. It has been carefully annotated by medical experts (experienced endoscopists). Endoscopy is the insertion of a long, thin tube directly into the body to observe an internal organ or tissue in detail, illustrated in figure 1. The dataset consists of 8000 images comprised of 8 classes showing anatomical landmarks, pathological findings, or endoscopic procedures in the GI tract. The dataset is considered to be balanced, given that there are 1000 images per class. One sample per each of the eight classes can be seen in figure 2. The dataset was split with an 80/20 ratio, where 80% (6400 samples) were used to train the models, and 20% (1600 samples) were retained as a test dataset to verify the model performance on unseen instances.

Three of the classes represent anatomical landmarks, three represent pathological states, and two are related to lesion-removal. The three anatomical landmarks are the pylorus, Z-line, and cecum, as seen in figure 1. A normal pylorus can be seen in figure 2e and is defined as the area around the opening from the stomach into the first part of the small bowel (see figure 1). Identifying the pylorus is essential when performing endoscopy of the duodenum, an important step in diagnosing celiac disease. A normal Z-line can be observed in figure 2f and is defined as the transition between the esophagus and the stomach (see figure 1).

The Z-line is an important anatomical landmark to examine as the disease esophagitis usually becomes evident at this location. A normal cecum

can be seen in figure 2d and is located close to the ileocecal valve (see figure 1). Reaching the cecum during an endoscopy marks the procedure's completion and is regarded as a quality indicator of the examination. The three classes representing pathological states, or more commonly denoted diseases, are esophagitis, ulcerative colitis, and polyps. A sample image of esophagitis can be seen in figure 2b and is caused by an inflammation of the esophagus. Ulcerative colitis is a disease that is characterized by inflammation of the large bowel, and a sample image can be seen in figure 2h.



(a) Dyed resection margins (b) Esophagitis

(c) Dyed lifted polyp (d) Normal cecum

(e) Normal pylorus (f) Normal Z-line

(g) Polyps (h) Ulcerative colitis

Figure 2: Samples from each class in Kvasir

As seen in figure 2g, polyps are outgrowths in the large bowel, and their identification and removal are essential to avoid cancer development. Lesion removal is the exercise of removing damaged tissue, and the two categories related to this are dyed lifted polyps (see figure 2c) and dyed resection margins (see figure 2a). When polyps are removed, the polyp is lifted and dyed to make the outgrowth more visible and easier to separate from the surrounding tissue. Medical images of resection margins are taken after polyp removal and are essential to ensure that all the damaged tissue was removed during the lesion removal [11, 44].

The images vary in capture angle, resolution, brightness, zoom, and center point. For the task of deep learning, the images are of reasonably high complexity, moderate quality, and relatively low volume [11]. Pogorelov et al. [44] consider the dataset big enough to perform image processing and machine learning with a relatively high accuracy [44]. However, it is seen that increasing the dataset size leads to improved performance, and the algorithms would likely perform better with an increased size of the dataset [60].

3.1 Preprocessing

The dataset images have different size dimensions, as illustrated in the samples, where figure 2b differs from the other samples. Dimensions in the dataset range from 720x576 up to 1920x1072 pixels, but the images are resized to the same resolution to fit the relevant method application during preprocessing. This step is essential as varying image resolution may hinder classification performance [29]. When extracting the raw pixels for KNN, all images are resized to 32x32 pixels. The reason for resizing the images to such a low resolution is that KNN has been seen to perform better on low-resolution images than high-resolution images [39]. When extracting the color histogram, the image resolution is not changed as the color histograms are dependent on resolution [49].

For CNN, the images are resized to 256x256 pixels. A CNN's performance is seen to improve with higher resolution as less of the details from the original image are lost [29]. However, increasing the resolution also increases how computationally expensive the model is, and 256x256 is considered a trade-off between resources and details. m

Medical images often contain unwanted artifacts, annotation information, or various markings [65], and the medical images in the Kvasir dataset is no exception. Examples of such artifacts are the green boxes seen in figures 2a and 2d and the annotation information seen in figures 2c, 2d, 2f and 2h. These disruptive elements may disturb the image classification and lead to false positives. Removal of such elements might improve the models' performance, and some of the studies conducted on the Kvasir dataset have had positive results when removing

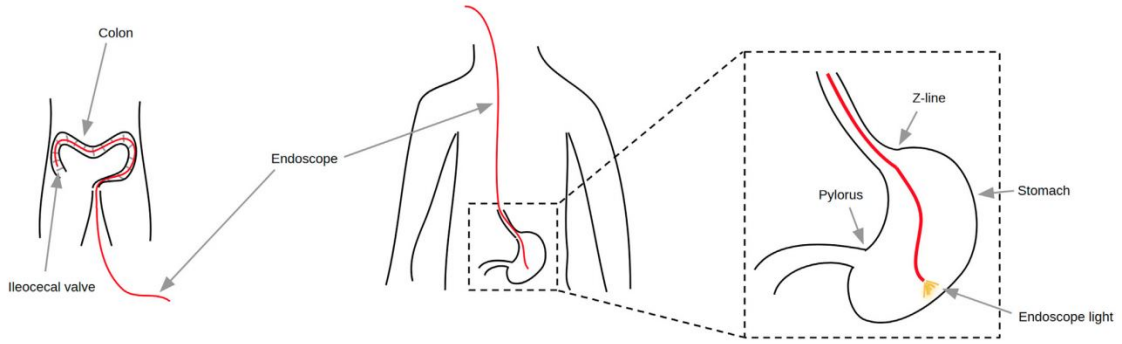


Figure 1: Endoscopy in the GI tract. Courtesy of Cogan et al. [11]

the artifacts [65, 31, 11]. However, removing the artifacts without losing information or generalizability is both a complex and time-consuming process. Thambawita et al. advise against removing the green boxes, and artifact removal has therefore not been done in this project [57].

3.2 Feature extraction

Feature engineering is the process of improving predictive modeling performance on datasets through data mining techniques that modify the data for better fitting in a specific ML method. When applying a ML method to a dataset, data samples or data points constitute the basic components. Every sample is described with several features, and every feature consists of different types of values. Feature extraction is a process where a new set of features can be created from the initial set while capturing all the essential information in a dataset [32]. This process is done to reduce the dimensionality of the data, where the dimensionality of the feature space is defined as the number of active features [55]. According to the *curse of dimensionality*, the more features we have, the more data we need to train a good model [36]. In the case of images, each pixel can be regarded as one feature. The images that are preprocessed for the CNN have a resolution of 256x256, which means 65536 features per image. When multiplying this with 8000 images, the total number of features is larger than 524 million, and dimensionality reduction is clearly necessary.

The CNN is a deep learning method defined as machine learning algorithms that use multiple layers to extract higher-level features from the input data progressively [52]. In practice, this means that when applying a deep learning algorithm on a dataset, the model can automatically learn features and perform feature extraction. Because of this, it is not necessary to perform feature engineering on the dataset prepared for the CNN. KNN is, in contrast, a traditional method where features have to be hand crafted [4]. We have performed two types of feature extraction for the KNN implementation; color histograms and pixel reduction. A color histogram is a representation of the color distribution in an image. An example histogram can be seen in figure 3. The histogram divides the color space into small intervals called bins, where each bin is one feature. There are three color channels (H, S, V) and eight bins per channel in this case. This process gives a total of 24 features. The second feature extraction method, pixel reduction, is done through the resizing in the preprocessing step. The images are resized to 32x32 pixels, which corresponds to 1024 pixels, and thereby 1024 features.

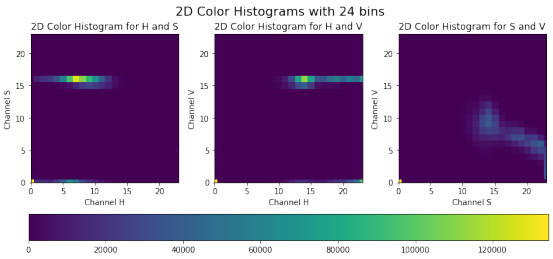


Figure 3: Color histogram of normal Z-line as seen in 2f.

3.3 Dataset bias

For a learning model to generalize and make predictions on unseen training data, it often makes certain assumptions based on the data. These assumptions are called the model's *inductive bias* [35]. The Kvasir dataset has been collected by Vestre Viken Hospital Trust, an organization that provides health services to people living in the geographical area of Vestre Viken.

As all images are captured by the same organization and in a limited geographical location, one can assume that the dataset has some capture bias level. Capture bias is inductive bias related to the acquisition of the images, both in terms of the used capturing device and the collector preferences for point of view, lighting conditions, and similar factors [58]. This bias could be related to the training the medical professionals have received or a limited number and variety of devices utilized. Also, all patients that have received endoscopic examinations live in Vestre Viken, and they are therefore exposed to the same environment, live in a similar cultural context, and to a large extent, have access to similar foods.

4 Methods

Two machine learning methods were implemented in this project; CNN and KNN. CNN is a powerful and versatile machine learning algorithm regarded as the popular approach to image recognition, segmentation, and classification [31]. Due to the high performance of CNNs in image classification, it is considered the state-of-the-art algorithm in computer-aided diagnostics [53, 56]. In contrast, KNN is a simple, traditional machine learning algorithm used for both regression and classification, which has been implemented in various fields due to its simplicity and explainability [36]. The explainability property of KNN makes the algorithm relevant for the medical field, as it is vital to establish trust when doctors plan to utilize predictions made by CAD in diagnostics. In addition, understanding of the algorithm enables doctors to apply domain expertise to find errors in the ML model predictions [47]. The methods used to implement both algorithms will be elaborated in this section, and how their performance was evaluated.

4.1 K-Nearest Neighbors

K-Nearest neighbor is a traditional supervised learning method used for both regression and classification [37]. KNN is based on instance-based learning, where the learning and generalization of the model are delayed until prediction time and not done preemptively on the training set as in CNN [36]. The algorithm learns and assigns a class for each query based on the most frequently occurring class in its K nearest samples. This assignment of labels is based on the assumption that K-nearest instances are similar in terms of their known attributes; they are likely to be similar in terms of their unknown attributes [26].

$$d_{minkowski} = \left(\sum_{i=1}^{n} (x_i - y_i)^p \right)^{\frac{1}{p}} \quad (1)$$

The KNN algorithm was implemented as an image classifier in this project. As mentioned in section 3.2, two feature extraction methods were applied; color histograms and pixel reduction. Two factors are important to measure the similarity or closeness of two instances; the number of neighbors, K, and the distance metric, d. The Minkowski distance, $d_{minkowski}$, is the generalized distance metric and can be seen in equation 1. The parameter p is a hyperparameter in equation 1. If $p = 1$, manhattan distance is used, and if $p = 2$, Euclidean distance is used [15]. Manhattan and Euclidean distance are two of the most commonly used distance metrics and are therefore evaluated in this project. The simple KNN has uniform weighting, where all the K nearest samples are equally treated when computing the class of the query. However, another variation of KNN, the weighted KNN, employs distance weighting, which gives additional weight to close neighbors according to their distances to the query instance [21].

Table 1: KNN grid search parameters

Parameter	Attribues
K	3, 5, 11, 19
Weighting	uniform, distance
Metric	Euclidean, Manhattan

K and p is the hyperparameters of KNN and can be optimized to improve the algorithm's performance through hyperparameter optimization (HPO) [15]. One of the most common HPO strategies is grid search, where a chosen set of values are optimized. In this case, the set of selected attributes are K, p, and weighting type, as seen in table 1. The attributes are utilized to form a set of trials by assembling every possible combination of the attributes [6]. Each trial was evaluated with 5-fold cross-validation. 5-fold cross-validation is an approach were cross-validation is performed 5 times with different partitionings of the data into training and validation sets, and the results are then averaged [36].

4.2 Convolutional Neural Networks

CNNs are a specialized kind of neural network. A neural network is a computational learning system that utilizes a network of functions that are built to understand and translate input data to the desired output data [36]. In ML, these functions are usually called *activation functions* and are often represented as neurons. This project utilizes the *rectified linear unit* (ReLU) activation function in all but the last layer, which is the standard activation function for deep neural networks as it avoids the *the vanishing gradient problem* [25, 42]. The equation for ReLU is shown in equation 2. The vanishing gradient problem may be encountered when training artificial neural networks (ANN) with gradient-based learning methods and backpropagation. Such ANNs update their weights based on the partial derivative of the loss function, and if the gradient becomes too small, training becomes hard [24]. The last layer's activation function is commonly the softmax function, which outputs a probability distribution over a finite set of outcomes [17, 63], meaning that the output is a probability that the input image belongs to each category.

$$f(x) = \begin{cases} x & \text{if x } \geq 0 \\ 0 & \text{if x} < 0 \end{cases} \tag{2}$$

The neurons are often placed sequentially in layers, as seen in figure 4. Each subsequent layer modifies the data in the hope of increasing the accuracy of the predictions by maximizing or minimizing its *objective function* to get an optimized output [3, 20]. In neural networks, the usual objective is to minimize the error, which is represented by a *loss function*. The cross-entropy loss function is commonly used for multi-class classification problems as it does not lead to saturation and slow learning in combination with softmax [46, 20].

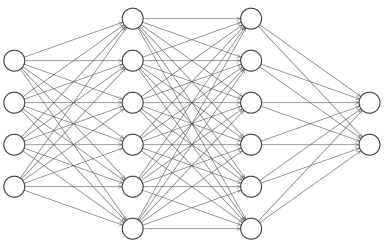


Figure 4: Example of neurons in a neural network

CNNs are a form of neural networks suited for processing data with a grid-like topology such as image data, which can be thought of as a 2-D grid of pixels [20]. The CNN architecture consists of several convolutional-pooling layer pairs followed by a fully connected network, as seen in figure 5. The convolutional layer is the core of a CNN and is characterized by a matrix, commonly known as a *filter*, that performs convolution on the input matrix. Convolution is a matrix transformation and is used to detect patterns in the input by taking the dot product of the matrix values covered by the filter. It is common to apply padding in the border of the input matrix. However, as the images in Kvasir naturally contains a black border without valuable information, padding is not used in this project. Pooling layers are commonly inserted between successive convolutional layers to reduce the number of parameters and computation in the network [20]. In this project, max-pooling with a filter size of 2x2 has been utilized, being one of the most common pooling-types and sizes [3]. Fully connected layers are layers where all neurons are connected with all the neurons of the subsequent layer [42]. The CNN feature extraction happens in the convolutional-pooling layers, while the fully connected network performs the actual classification [20].

4.2.1 Traditional convolution neural network

A CNN was constructed with architecture, as seen in figure 5, and with properties as in table 2. The height and the width correspond to the dimensions of the image matrix. For example, the CNN's input image has a resolution of 256x256, and the height and width of layer nr. 1 is (256-2). The subtraction of 2 is due to the absence of added padding. The depth corresponds to the number of filters applied in the layer, where all filters have the dimensions *filter height* x *filter width*. Layers 6-8 are fully connected.

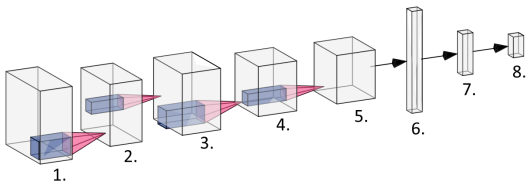


Figure 5: Implemented CNN architecture

Table 2: Parameters of implemented CNN

Layer number	Depth	Height	Width	Filter height	Filter width
1	32	254	254	3	3
2	32	127	127	2	2
3	64	125	125	3	3
4	64	62	62	2	2
5	64	60	60		
6	1	230400	1		
7	1	64	1		
8	1	8	1		

4.2.2 ResNet50

The second CNN model implemented was a ResNet50 convolutional neural network, a residual network (ResNet) with 50 layers. Residual networks are a variation of neural networks that enables the development of deep networks without the *degradation problem*. The degradation problem is observed when deepening a neural network increases the accuracy until it converges towards a limit value and then starts to decrease [22]. Residual networks were developed to remedy this problem and have promising results in medical image classification [30, 45].

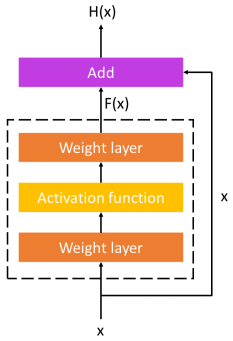
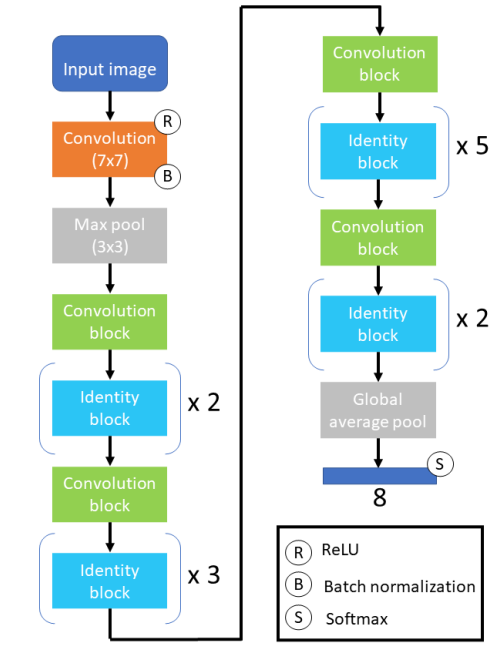
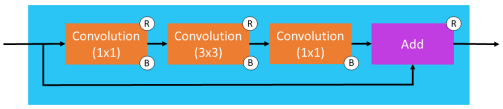


Figure 6: A residual block

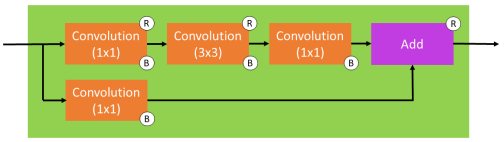
Residual networks consist of several residual blocks, as seen in figure 6. Each residual block can be represented by the function $H(x) = F(x) + x$, where $H(x)$ is the desired underlying mapping. $F(x)$ is the residual mapping and is the output of the stacked layers. x is the identity mapping and is added to $F(x)$ to form $H(x)$ through shortcut connections that skip some layers. These residual blocks ensure that deeper counterparts to shallower networks can achieve at least the same accuracy as their shallower counterparts, thus avoiding the degradation problem [22].



(a) ResNet50 overview



(b) Identity block



(c) Convolutional block

Figure 7: Architecture of ResNet50

A single network can use different residual blocks, and optimal placing and tuning can vary from problem to problem. Several beneficial network designs have been created, and we have implemented one of them, the ResNet50 model, a design optimized for image classification. The architecture of ResNet50 can be seen in figure 7a. With the first 7x7 convolutional layer in the beginning and the final fully-connected layer, there are 50 layers in total. There are two kinds of residual blocks implemented in ResNet50; the identity block (as seen in figure 7b) and the convolutional block (as seen in figure 7c). The convolutional block differs from the identity block due to the convolutional layer at the short cut, which the identity block does not have. Both blocks have 1x1 convolutional layers added to

the start and end of the networks. This technique, called bottleneck design, reduces the parameter amount without too much performance degradation [28]. The activation function used is ReLU, and batch normalization is performed for each convolutional layer. Batch normalization normalizes a previous activation layer's output by subtracting the batch mean and dividing by the batch standard deviation. This is done to increase the stability and efficiency of a neural network [27].

As mentioned in section 3, the Kvasir dataset is relatively low in volume. When the volume is low, there is a risk of overfitting due to insufficient training data. As a way to avoid overfitting during training, 50% *dropout* was performed. Dropout is a method where random neurons are deactivated to reduce overfitting due to noise in the training data [54].

In addition to performing dropout, *transfer learning* was utilized to improve the model's generalization. In transfer learning, generalized data from similar, pre-trained models are utilized to improve performance. By using generalized weights from models with good performance, the network has a better starting point than if it was initialized with random weights [64]. In this model, transfer learning was done using weights pre-trained on the ImageNet competition database [48]. The database has more than 14 million images that are manually annotated to more than 20 000 different classes, and it is regarded as the standard benchmark for large-scale object recognition [62, 48].

4.3 Evaluation Metrics

When evaluating a classification model's performance, a confusion matrix can be created based on the correctly and wrongly predicted labels. A confusion matrix is a good method to measure the precision and recall scores of the model's predictions. For each class, the matrix displays the number of right and wrong, which gives information on how well the model predicts specific classes. A confusion matrix for a binary classification problem can be seen in table 3. The horizontal axis corresponds to predicted labels, and the vertical axis corresponds to the correct labels.

Table 3: Example of a binary classification confusion matrix

	0	**1**
0	True Negative	False Positive
1	False Negative	True Positive

True negatives and true positives represent correct predictions by the model, and it is beneficial to have as many of these as possible. Typically, one would have to consider the impact of false negatives and false positives concerning the given problem. When it comes to medical diagnosis, false positives and false negatives might be critical as it may lead to patients being treated based on false assumptions.

There are various metrics to evaluate a model's performance, and some of the most common ones are accuracy, precision, recall, and F1 score. Accuracy is defined as the ratio of correctly predicted labels to the total number of predictions, as seen in equation 3. This metric is used to evaluate the overall performance of the model. Precision is the ratio of correctly predicted positive labels to the total predicted positive labels, as seen in equation 4). The recall is the ratio of correctly predicted positive labels to all positive labels in the test data, as seen in equation 5. Higher values of these metrics indicate good performance of the model [40]. The F1 score is a measure of the balance between the precision and the recall, as seen in equation 6 [50].

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{3}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{4}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{5}$$

$$\text{F1 score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

5 Results

In the following section, the project experiments' results are presented before comparing the different approaches towards the end of the chapter.

5.1 K-Nearest Neighbors

In this project, we implemented two variations of K-Nearest Neighbor algorithms using SciKit-Learn [41]. The first implementation used the input image's pixel intensities as features, while the second one used the color distribution in the input images, as described in section 4.1.

The first model was based on pixel intensity in resized 32x32 pixel input images. It was found to have the highest performance with 11 neighbors, Euclidean distance, and a distance-based weighting function, as seen in the search results found in table 9 in appendix A. An attempt was also made to increase the image size to 100x100 pixels; however, this only increased the inference time without significantly affecting the model performance.

When used to predict instances on the unseen testing part of the dataset, the top-ranking model from the grid search achieved an overall accuracy score of 65%. The results of the predictions compared to their true labels are shown in the confusion matrix in figure 12 in appendix C, and summarized in the classification report in table 4. They indicate that the model performs better on the anatomical landmark classes normal cecum and normal pylorus, while the detection of polyps, esophagitis and dyed lifted polyps are inconsistent.

The second model implementation was a KNN classifier using color histograms of the input images as features. After performing a grid search, the top-ranking model hyperparameters were found to be 5 number of neighbors, Manhattan distance, and a distance-based weighting function, as seen in table 10 in appendix B.

Using the color histogram-based KNN model with the top-ranking hyperparameters to predict instances on the unseen test data, the model achieved an overall accuracy score of 66%. This score is an improvement compared to the first model, which achieved a score of 65%. The results of the predictions compared to their true labels are shown in the confusion matrix in figure 8, with the results summarized in the classification report in table 5. As seen in these evaluations, the model still suffers from the same inconsistencies as the first model. Interestingly, compared to the first model, this model is observed to perform better on the classes where the color might play a distinct role in classification, such as dyed lifted polyps, dyed resection margins, and polyps.

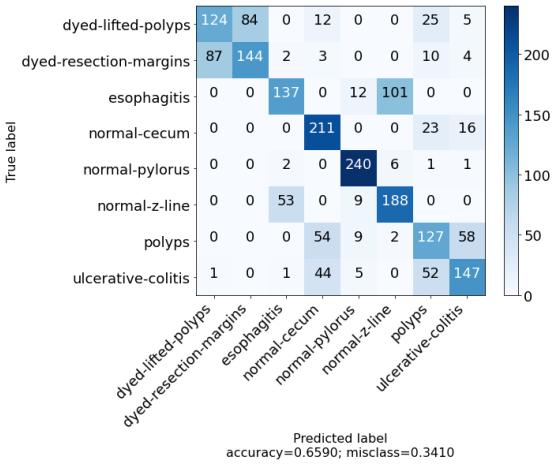


Figure 8: Confusion matrix for KNN model based on color histograms

5.2 Convolutional Neural Networks

In addition to the two K-Nearest Neighbor implementations, we implemented two convolutional neural network variations using Keras [9], as described in section 4.2. The first implementation was a traditional CNN architecture, described in section 4.2.1. The second model was a residual neural network, ResNet50, using pre-trained weights on the ImageNet competition database. After training the first CNN model for 30 epochs on the training dataset, it achieved an accuracy score of 75% when predicting instances on the unseen testing dataset.

The results of the predictions compared to their correct labels are shown in the confusion matrix in figure 13 in appendix D, and summarized in the classification report in table 6. These confirm that the model has some of the same inconsistencies in its predictions as the previously implemented traditional models.

The second model was implemented and tuned in two iterations. In the first iteration, the model and its pre-trained weights from ImageNet were compiled as-is, and layers were frozen to avoid excessive training. The model's top layers were replaced with our sequence of a pooling, dropout, and fully connected layer fit to our problem task before training these layers. In the second iteration, all the model layers were unfrozen and made trainable again, and trained further with a lower learning rate in an attempt to fine-tune the network even more.

After training the first iteration of the model for

Table 4: Classification report for KNN model based on raw pixel intensities

	precision	recall	f1-score	support
dyed-lifted-polyps	0.59	0.24	0.34	200.0
dyed-resection-margins	0.6	0.69	0.64	200.0
esophagitis	0.8	0.55	0.65	200.0
normal-cecum	0.59	0.94	0.73	200.0
normal-pylorus	0.85	0.88	0.87	200.0
normal-z-line	0.59	0.82	0.68	200.0
polyps	0.51	0.38	0.43	200.0
ulcerative-colitis	0.66	0.69	0.67	200.0
accuracy	0.65	0.65	0.65	0.65
macro avg	0.65	0.65	0.63	1600.0
weighted avg	0.65	0.65	0.63	1600.0

Table 5: Classification report for KNN model based on color histograms

	precision	recall	f1-score	support
dyed-lifted-polyps	0.58	0.5	0.54	250.0
dyed-resection-margins	0.63	0.58	0.6	250.0
esophagitis	0.7	0.55	0.62	250.0
normal-cecum	0.65	0.84	0.74	250.0
normal-pylorus	0.87	0.96	0.91	250.0
normal-z-line	0.63	0.75	0.69	250.0
polyps	0.53	0.51	0.52	250.0
ulcerative-colitis	0.64	0.59	0.61	250.0
accuracy	0.66	0.66	0.66	0.66
macro avg	0.66	0.66	0.65	2000.0
weighted avg	0.66	0.66	0.65	2000.0

30 epochs, it achieved an overall accuracy score of 91% when predicting instances on the unseen test dataset. The results of the predictions compared to their true labels are shown in the confusion matrix in figure 14 in appendix E, and summarized in the classification report in table 11 in appendix F. They show that the model performed significantly better than the previous models. However, the model is still struggling to differentiate between some classes and struggles with esophagitis and normal Z-line in particular.

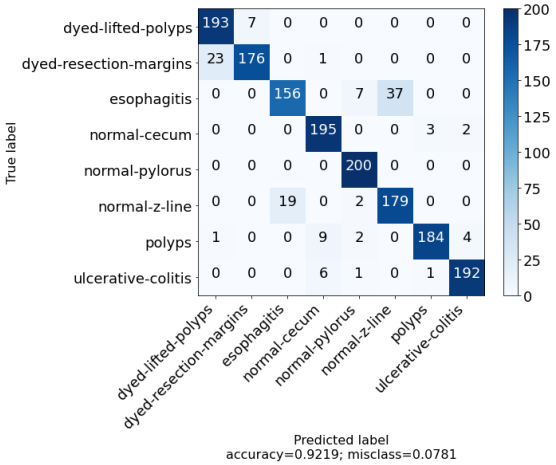


Figure 9: Confusion matrix for the final tuned ResNet CNN

In the second iteration of the model, the previously frozen layers were unfrozen. It was trained further for 30 epochs with all layers made trainable and with a lower learning rate to see if the model could gain any improvements. When making predictions on the test set, this model achieved a 92% accuracy score, improving one percentage point from the previous iteration.

The results of the predictions compared to their true labels are shown in the confusion matrix in figure 9, and summarized in the classification report in table 7. The results show the model improved its overall performance. The problems with the classification of esophagitis and normal Z-line were reduced but are still present.

5.3 Results summary

After experimenting with five variations of two machine learning approaches for image classification, a significant increase in performance can be seen from the simple, traditional model to state-of-the-art deep learning techniques. As seen in table 8, the best ranking model achieves an accuracy score of over 92%, while the simpler models score between 65% and 66%.

Table 8: Classification report summary for all trained models. From left to right: Pixel intensity KNN, color histogram KNN, CNN, first ResNet50 iteration, fine-tuned ResNet50 iteration

	KNN 1	KNN 2	CNN	ResNet50	ResNet50-2
accuracy	0.65	0.66	0.75	0.91	0.92
precision macro avg	0.65	0.66	0.75	0.91	0.92
precision weighted avg	0.65	0.66	0.75	0.91	0.92
recall macro avg	0.65	0.66	0.75	0.91	0.92
recall weighted avg	0.65	0.66	0.75	0.91	0.92
f1-score macro avg	0.63	0.65	0.74	0.91	0.92
f1-score weighted avg	0.63	0.65	0.74	0.91	0.92

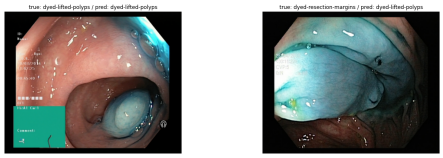
It is clear that for a KNN model, capturing enough important information in either 32x32 or 100x100 pixel images was not possible. They scored well mostly on the more straightforward landmark classes. The first CNN implementation showed that it could generalize better than the KNN even with a shallow layer structure and little parameter tuning. In contrast, the ResNet implementation outperformed all the other models used in the project.

Table 6: Classification report for CNN implementation

	precision	recall	f1-score	support
dyed-lifted-polyps	0.66	0.6	0.63	200.0
dyed-resection-margins	0.65	0.71	0.68	200.0
esophagitis	0.86	0.52	0.65	200.0
normal-cecum	0.78	0.94	0.85	200.0
normal-pylorus	0.87	0.97	0.92	200.0
normal-z-line	0.67	0.86	0.76	200.0
polyps	0.77	0.66	0.71	200.0
ulcerative-colitis	0.79	0.74	0.76	200.0
accuracy	0.75	0.75	0.75	0.75
macro avg	0.76	0.75	0.74	1600.0
weighted avg	0.76	0.75	0.74	1600.0

Table 7: Classification report for the final tuned ResNet CNN

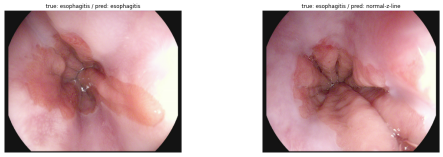
	precision	recall	f1-score	support
dyed-lifted-polyps	0.89	0.96	0.93	200.0
dyed-resection-margins	0.96	0.88	0.92	200.0
esophagitis	0.89	0.78	0.83	200.0
normal-cecum	0.92	0.98	0.95	200.0
normal-pylorus	0.94	1.0	0.97	200.0
normal-z-line	0.83	0.9	0.86	200.0
polyps	0.98	0.92	0.95	200.0
ulcerative-colitis	0.97	0.96	0.96	200.0
accuracy	0.92	0.92	0.92	0.92
macro avg	0.92	0.92	0.92	1600.0
weighted avg	0.92	0.92	0.92	1600.0



(a) Correctly predicted dyed lifted polyps as dyed lifted polyps

(b) Incorrectly predicted dyed lifted polyps as dyed resection margins

Figure 10: Sample classifications for dyed lifted polyps and dyed resection margins



(a) Correctly predicted esophagitis as esophagitis

(b) Incorrectly predicted esophagitis as normal z line

Figure 11: Sample classifications for esophagitis and normal z line

On a class-by-class basis, none of the models performed over 90% for all classes. Interestingly, most of the models struggled with the same classes with varying degrees of error. When examining the misclassifications and comparing it to correct examples, it is apparent that many of the wrongly classified predictions happen between two visually similar classes. Examples include incorrect predictions between esophagitis and normal z line, dyed lifted polyps and dyed resection margins, and ulcerative colitis and polyps. Two examples of this is seen in figure 10 and 11.

6 Conclusion

This project's main goal was to compare and perform image classification of images from the GI tract using KNN, a traditional machine learning approach, and CNN, the modern, state-of-the-art approach to classification tasks. A total of five variations of these image classification models were implemented and tested against the Kvasir dataset, consisting of images of anatomical landmarks, pathological states, and endoscopic procedures in the GI tract.

The top-performing model was a ResNet-based CNN model, with weights pre-trained on the ImageNet competition database. Unsurprisingly, this CNN approach outperformed the KNN implementations in our experiments, achieving a 92% accuracy score when tested against unseen instances in the dataset. In comparison, medical experts in the field achieve a diagnostic accuracy of 82.4–87.3% [53]. The KNN model underperformed both compared to medical experts and CNN, reaching a maximum accuracy score of 66%. It was seen that both the CNN and the KNN struggled to predict samples from the same classes where they are visually similar to a varying extent.

Further work could be refining the data preprocessing. Some images contain unwanted artifacts such as black frames, green markings, and text, which possibly are affecting the learning task. Developing a data preprocessing that efficiently removes the unwanted artifacts without a significant loss of information would therefore be beneficial. In addition, the ResNet50 could have been made deeper. The deeper residual networks evaluated by Kaiming et al. in *Deep residual learning for image recognition* performed better compared to its shallower counterparts, and investigating if the performance would increase would therefore be interesting.

Bibliography

[1] Taruna Agrawal, Rahul Gupta, and Shrikanth Narayanan. On evaluating CNN representations for low resource medical image classification. *arXiv e-prints*, page arXiv:1903.11176, March 2019.

[2] Jamil Ahmad, Khan Muhammad, Mi Lee, and Sung Baik. Endoscopic image classification and retrieval using clustered convolutional features. *Journal of Medical Systems*, 41, 12 2017.

[3] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*. IEEE, August 2017.

[4] Syed Anwar, Muhammad Majid, Adnan Qayyum, Muhammad Awais, Majdi Alnowami, and Khurram Khan. Medical image analysis using convolutional neural networks: A review. *Journal of Medical Systems*, 42:226, 10 2018.

[5] Hiba Asri, Hajar Mousannif, Hassan Al Moatassime, and Thomas Noel. Using machine learning algorithms for breast cancer risk prediction and diagnosis. *Procedia Computer Science*, 83:1064 – 1069, 2016. The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016) / The 6th International Conference on Sustainable Energy Information Technology (SEIT-2016) / Affiliated Workshops.

[6] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13(null):281–305, February 2012.

[7] Rune Johan Borgli, Hakon Kvale Stensland, Michael Alexander Riegler, and Pal Halvorsen. Automatic hyperparameter optimization for transfer learning on medical image datasets using bayesian optimization. In *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*. IEEE, May 2019.

[8] Hermann Brenner, Matthias Kloor, and Christian Peter Pox. Colorectal cancer. *The Lancet*, 383(9927):1490 – 1502, 2014.

[9] François Chollet. Keras. `https://github.com/fchollet/keras`, 2015.

[10] Adrienne Chu, Hongshik Ahn, Bhawna Halwan, Bruce Kalmin, Everson L.A. Artifon, Alan Barkun, Michail G. Lagoudakis, and Atul Kumar. A decision support system to facilitate management of patients with acute gastrointestinal bleeding. *Artificial Intelligence in Medicine*, 42(3):247 – 259, 2008.

[11] Timothy Cogan, Maribeth Cogan, and Lakshman Tamil. Mapgi: Accurate identification of anatomical landmarks and diseased tissue in gastrointestinal tract using deep learning. *Computers in Biology and Medicine*, 111:103351, 2019.

[12] Thomas de Lange, Pål Halvorsen, and Michael Riegler. Methodology to develop machine learning algorithms to improve performance in gastrointestinal endoscopy. *World Journal of Gastroenterology*, 24, 12 2018.

[13] Thomas de Lange, Pål Halvorsen, and Michael Riegler. Methodology to develop machine learning algorithms to improve performance in gastrointestinal endoscopy. *World Journal of Gastroenterology*, 24, 12 2018.

[14] Dayna S. Early, Tamir Ben-Menachem, G. Anton Decker, John A. Evans, Robert D. Fanelli, Deborah A. Fisher, Norio Fukami, Joo Ha Hwang, Rajeev Jain, Terry L. Jue, Khalid M. Khan, Phyllis M. Malpas, John T. Maple, Ravi S. Sharaf, Jason A. Dominitz, and Brooks D. Cash. Appropriate use of gi endoscopy. *Gastrointestinal Endoscopy*, 75(6):1127 – 1131, 2012.

[15] Matthias Feurer and Frank Hutter. Hyperparameter optimization. In *Automated Machine Learning*, pages 3–33. Springer, Cham, 2019.

[16] Chathurika Gamage, Isuru Wijesinghe, Charith Chitraranjan, and Indika Perera. GI-net: Anomalies classification in gastrointestinal tract through endoscopic imagery with deep learning. In *2019 Moratuwa Engineering Research Conference (MERCon)*. IEEE, July 2019.

[17] Bolin Gao and Lacra Pavel. On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv:1704.00805*, 2017.

[18] N. Ghatwary, X. Ye, and M. Zolgharni. Esophageal abnormality detection using densenet based faster r-cnn with gabor features. *IEEE Access*, 7:84374–84385, 2019.

[19] Rafael C. Gonzalez and Richard E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J., 2008.

[20] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[21] Jianping Gou, Lan Du, Yuhong Zhang, Taisong Xiong, et al. A new distance-weighted k-nearest neighbor classifier. *J. Inf. Comput. Sci*, 9(6):1429–1436, 2012.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[23] David Hewett, Charles Kahi, and Douglas Rex. Efficacy and effectiveness of colonoscopy: How do we bridge the gap? *Gastrointestinal endoscopy clinics of North America*, 20:673–84, 10 2010.

[24] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6:107–116, 04 1998.

[25] Hidenori Ide and Takio Kurita. Improvement of learning for CNN with ReLU activation by sparse regularization. In *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, May 2017.

[26] Sadegh Bafandeh Imandoust and Mohammad Bolandraftar. Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background. *S B Imandoust et al. Int. Journal of Engineering Research and Applications*, 3(5):605–609, 2013.

[27] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*, 2015.

[28] Qingge Ji, Jie Huang, Wenjie He, and Yankui Sun. Optimized deep convolutional neural networks for identification of macular diseases from optical coherence tomography images. *Algorithms*, 12:51, 02 2019.

[29] Suresh Kannojia and Gaurav Jaiswal. Effects of varying resolution on performance of cnn based image classification an experimental study. *International Journal of Computer Sciences and Engineering*, 6:451–456, 09 2018.

[30] Nitish Shirish Keskar and Richard Socher. Improving generalization performance by switching from adam to SGD. *CoRR*, abs/1712.07628, 2017.

[31] M. Kirkerød, R. J. Borgli, V. Thambawita, S. Hicks, M. A. Riegler, and P. Halvorsen. Unsupervised preprocessing to improve generalisation for medical image classification. In *2019 13th International Symposium on Medical Information and Communication Technology (ISMICT)*, pages 1–6, 2019.

[32] Konstantina Kourou, Themis P. Exarchos, Konstantinos P. Exarchos, Michalis V. Karamouzis, and Dimitrios I. Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13:8 – 17, 2015.

[33] Faisal Mahmood, Ziyun Yang, Thomas Ashley, and Nicholas J. Durr. Multimodal Densenet. *arXiv e-prints*, November 2018.

[34] Jun Min, Min Kwak, and Jae Cha. Overview of deep learning in gastrointestinal endoscopy. *Gut and Liver*, 13, 01 2019.

[35] Tom M. Mitchell. The need for biases in learning generalizations. In *Readings in Machine Learning*. Morgan Kauffman, 1980.

[36] Tom M. Mitchell. *Machine Learning*. McGraw-Hill Education, 1997.

[37] Mohssen Mohammed, Muhammad Badruddin Khan, and Eihab Bashier Mohammed Bashier. *Machine learning: algorithms and applications*. CRC Press, Taylor & Francis Group, Boca Raton, 2017.

[38] Abdullah Nahid and Yinan Kong. Involvement of machine learning for breast cancer image classification: A survey. *Computational and Mathematical Methods in Medicine*, 2017:1–29, 12 2017.

[39] Iva Nurwauziyah, Umroh Sulistyah, I Gede, I Gede Brawiswa Putra, and Muhammad Firdaus. Satellite image classification using decision tree, svm and k-nearest neighbor. 07 2018.

[40] David Olson. *Advanced data mining techniques*. Springer, Berlin, 2008.

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[42] Francisco Câmara Pereira and Stanislav S. Borysov. Chapter 2 - machine learning fundamentals. In Constantinos Antoniou, Loukas Dimitriou, and Francisco Pereira, editors, *Mobility Patterns, Big Data and Transport Analytics*, pages 9 – 29. Elsevier, 2019.

[43] Konstantin Pogorelov, O. Ostroukhova, Mattis Jeppsson, Håvard Espeland, C. Griwodz, T. D. Lange, D. Johansen, M. Riegler, and P. Halvorsen. Deep learning and hand-crafted feature based approaches for polyp detection in medical videos. *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*, pages 381–386, 2018.

[44] Konstantin Pogorelov, Kristin Ranheim Randel, Carsten Griwodz, Sigrun Losada Eskeland, Thomas de Lange, Dag Johansen, Concetto Spampinato, Duc-Tien Dang-Nguyen, Mathias Lux, Peter Thelin Schmidt, Michael Riegler, and Pål Halvorsen. Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection. In *Proceedings of the 8th ACM on Multimedia Systems Conference*, MMSys'17, pages 164–169, New York, NY, USA, 2017. ACM.

[45] A. S. B. Reddy and D. S. Juliet. Transfer learning with resnet-50 for malaria cell-image classification. In *2019 International Conference on Communication and Signal Processing (ICCSP)*, pages 0945–0949, 2019.

[46] R. Reed and R.J. MarksII. *Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks*. A Bradford Book. MIT Press, 1999.

[47] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[48] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115, 09 2014.

[49] S. Sablak and T. Boult. Multilevel color histogram representation of color images by peaks for omni-camera. In *SIP*, 1999.

[50] Yutaka Sasaki. The truth of the f-measure. *Teach Tutor Mater*, 01 2007.

[51] Li Q Shiraishi J, Appelbaum D, and Doi K. Computer-aided diagnosis and artificial intelligence in clinical imaging. *Semin Nucl Med.*, 41(6):449–462, 2011.

[52] R. Sinha, R. Pandey, and R. Pattnaik. Deep learning for computer vision tasks: A review. *ArXiv*, abs/1804.03928, 2018.

[53] Eun Song, Beomhee Park, Chun-Ae Ha, Sung Wook Hwang, sang hyoung Park, Dong-Hoon Yang, Byong Ye, Seung-Jae Myung, Suk-Kyun Yang, Namkug Kim, and Rupert Leong. Endoscopic diagnosis and treatment planning for colorectal polyps using a deep-learning model. *Scientific Reports*, 10:30, 12 2020.

[54] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January 2014.

[55] M. Steinbach T. Pang-Ning and V. Kumar. *Introduction to data mining*, 2006.

[56] Hirotoshi Takiyama, Tsuyoshi Ozawa, Soichiro Ishihara, Mitsuhiro Fujishiro, Satoki Shichijo, Shuhei Nomura, Motoi Miura, and Tomohiro Tada. Automatic anatomical classification of esophagogastroduodenoscopy images using deep convolutional neural networks. *Scientific Reports*, 8, 05 2018.

[57] Vajira Thambawita, Debesh Jha, Hugo Hammer, Håvard Johansen, Dag Johansen, Pål Halvorsen, and Michael Riegler. An extensive study on cross-dataset bias and evaluation metrics interpretation for machine learning applied to gastrointestinal tract abnormality classification. *ACM Transactions on Computing for Healthcare*, 1:1–29, 06 2020.

[58] Tatiana Tommasi, Novi Patricia, Barbara Caputo, and Tinne Tuytelaars. *A Deeper Look at Dataset Bias*, pages 37–55. Springer International Publishing, Cham, 2017.

[59] Pu Wang, Xiao Xiao, Jeremy Brown, Tyler Berzin, Mengtian Tu, Fei Xiong, Xiao Hu, Peixi Liu, Yan Song, Di Zhang, Xue Yang, Lianping Li, Jiong He, Xin Yi, Jingjia Liu, Xiaogang Liu, and Lucinda Lai. Development and validation of a deep-learning algorithm for the detection of polyps during colonoscopy. *Nature Biotechnology*, pages 741–748, 10 2018.

[60] Philip Woodall, Alexander Borek, Jing Gao, Martin Oberhofer, and Andy Koronios. An investigation of how data quality is affected by dataset size in the context of big data analytics. 08 2014.

[61] Juri Yanase and Evangelos Triantaphyllou. A systematic survey of computer-aided diagnosis in medicine: Past and present developments. *Expert Systems with Applications*, 138:112821, 07 2019.

[62] M. Zabir, N. Fazira, Zaidah Ibrahim, and Nurbaity Sabri. Evaluation of pre-trained convolutional neural network models for object recognition. *International Journal of Engineering and Technology (UAE)*, 7:95–98, 08 2018.

[63] Qiuyu Zhu, Zikuang He, Tao Zhang, and Wennan Cui. Improving classification performance of softmax loss function based on scalable batch-normalization. *Applied Sciences*, 10:2950, 04 2020.

[64] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, pages 1–34, 2020.

[65] Şaban Öztürk and Umut Özkaya. Gastrointestinal tract classification using improved lstm based cnn. *Multimedia Tools and Applications*, 79, 10 2020.

Appendix

A Grid search results for KNN model based on pixel intensities

Table 9: Grid search results for KNN model based on pixel intensities

rank	iteration	param-metric	param-n-neighbors	param-weights	mean-test-score	std-test-score
1	5	euclidean	11	distance	0.6456	0.0105
2	4	euclidean	11	uniform	0.645	0.011
3	3	euclidean	5	distance	0.6428	0.0109
4	13	manhattan	11	distance	0.6427	0.0097
5	7	euclidean	19	distance	0.6383	0.0139
6	12	manhattan	11	uniform	0.6377	0.0064
7	2	euclidean	5	uniform	0.6372	0.0089
8	6	euclidean	19	uniform	0.6369	0.015
9	15	manhattan	19	distance	0.6362	0.0048
10	11	manhattan	5	distance	0.6344	0.0165
11	8	manhattan	3	uniform	0.6331	0.0064
12	10	manhattan	5	uniform	0.6319	0.0171
13	14	manhattan	19	uniform	0.6314	0.0034
14	9	manhattan	3	distance	0.63	0.0049
15	0	euclidean	3	uniform	0.6242	0.0045
16	1	euclidean	3	distance	0.623	0.0072

B Grid search results for KNN model based on color histograms

Table 10: Grid search results for KNN model based on color histograms

rank	iteration	param-metric	param-n-neighbors	param-weights	mean-test-score	std-test-score
1	11	manhattan	5	distance	0.6777	0.0122
2	13	manhattan	11	distance	0.6763	0.0093
3	15	manhattan	19	distance	0.676	0.0092
4	9	manhattan	3	distance	0.6742	0.0111
5	10	manhattan	5	uniform	0.6737	0.0153
6	8	manhattan	3	uniform	0.6722	0.0134
7	12	manhattan	11	uniform	0.6712	0.0082
8	14	manhattan	19	uniform	0.6708	0.0107
9	7	euclidean	19	distance	0.6615	0.0176
10	5	euclidean	11	distance	0.6582	0.0154
11	6	euclidean	19	uniform	0.656	0.0191
12	3	euclidean	5	distance	0.653	0.0181
13	2	euclidean	5	uniform	0.6505	0.0193
14	4	euclidean	11	uniform	0.6475	0.0145
15	1	euclidean	3	distance	0.6395	0.0154
16	0	euclidean	3	uniform	0.6362	0.0143

C Confusion matrix for KNN model based on pixel intensities

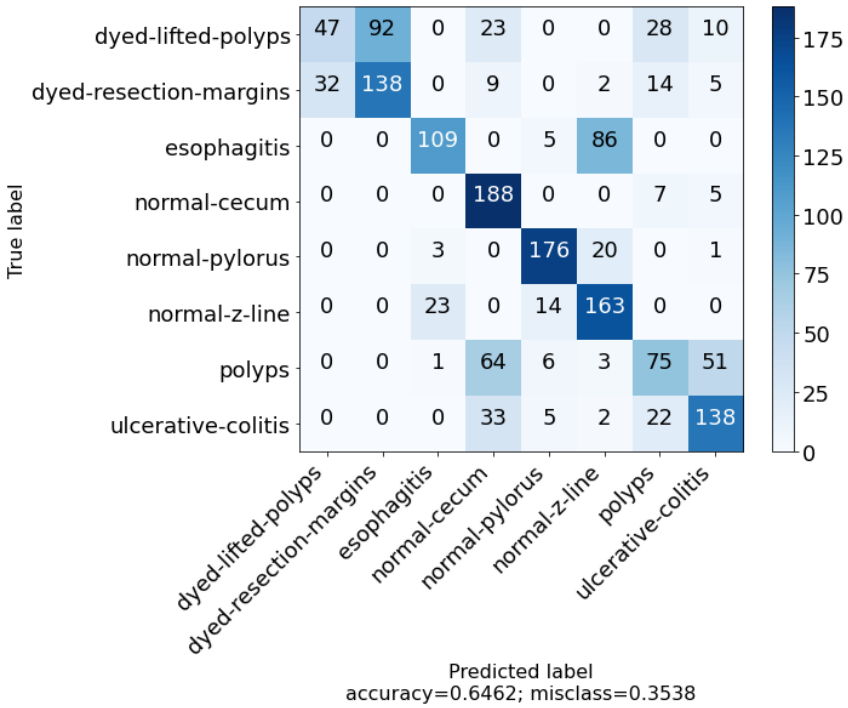


Figure 12: Confusion matrix for KNN model based on pixel intensities

D Confusion matrix for CNN implementation

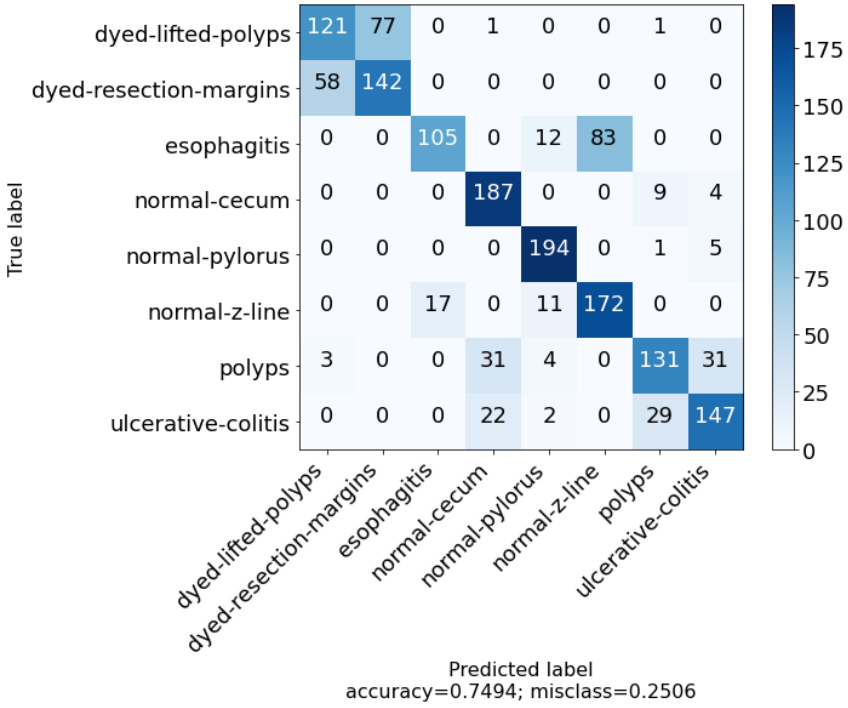


Figure 13: Confusion matrix for CNN implementation

E Classification report for ResNet CNN

Table 11: Classification report for ResNet CNN

	precision	recall	f1-score	support
dyed-lifted-polyps	0.88	0.94	0.91	200.0
dyed-resection-margins	0.93	0.87	0.9	200.0
esophagitis	0.88	0.77	0.82	200.0
normal-cecum	0.92	0.97	0.94	200.0
normal-pylorus	0.94	1.0	0.97	200.0
normal-z-line	0.81	0.88	0.84	200.0
polyps	0.97	0.92	0.94	200.0
ulcerative-colitis	0.97	0.96	0.96	200.0
accuracy	0.91	0.91	0.91	0.91
macro avg	0.91	0.91	0.91	1600.0
weighted avg	0.91	0.91	0.91	1600.0

F Confusion matrix for ResNet CNN

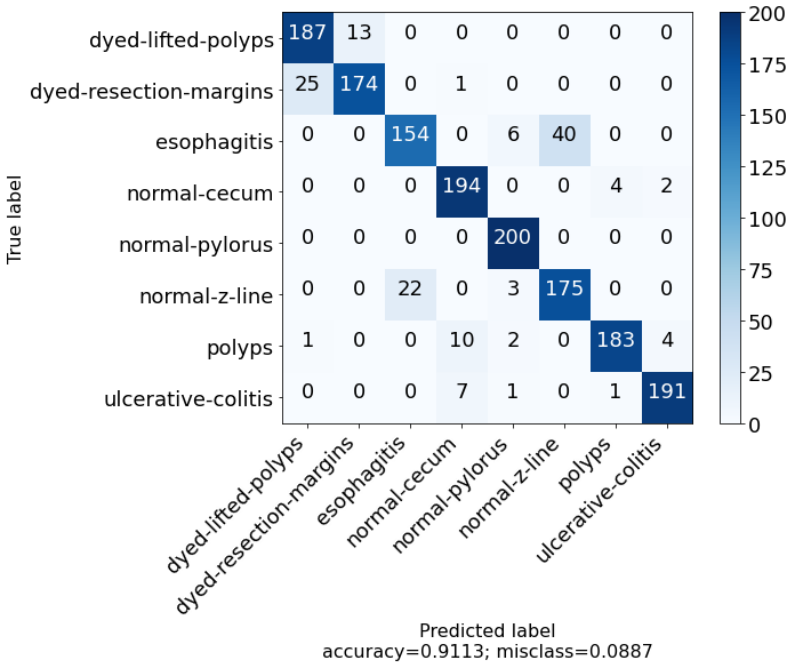


Figure 14: Confusion matrix for ResNet CNN