Classification of Porosity in Sulfur-Based Concrete Samples Using Deep Neural Networks

Alireza Behravesh

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Technology
in
Information Technology

Eastern Mediterranean University
September 2023
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Ali Hakan Ulusoy
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Technology in Information Technology.

Asst. Prof. Dr. Ece Çelik
Director, School of Computing and Technology

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Technology in Information Technology.

Prof. Dr. Ahmet Rizaner
Supervisor

Examining Committee

1. Prof. Dr. Ahmet Rizaner

2. Assoc. Prof. Dr. Kamil Yurtkan

3. Asst. Prof. Dr. Hüsnü Bayramoğlu

ABSTRACT

This study investigates the application of deep learning and machine learning techniques for the classification of sulfur-based concrete samples based on porosity, which is an important property that affects the strength, durability, and permeability of concrete.

The first part of the research focused on creating a unique dataset of sulfur-based concrete samples and calculating features such as porosity. The percentage porosity was then calculated, and images were labeled as low porosity or high porosity based on the percentage of porosity. The images of physical samples were automatically annotated by image processing techniques to create a dataset.

The second part of the study aimed to train and test a neural network to predict and classify samples based on porosity. We classified concrete images into two separate classes of low and high porosity using a basic Convolutional Neural Network (CNN) and transfer learning with a pre-trained model such as AlexNet. Porosity was calculated as the distribution of air voids and aggregates through the concrete sample. The comparison of two of the best models and finding the accuracy and other performance metrices of the networks were done using confusion matrices.

The conclusion of this study shows that pre-trained models with transfer learning, such as AlexNet, can be used to accurately and automatically classify sulfur-based concrete samples based on porosity, which could lead to faster and more efficient quality control of concrete production. This study also sets the stage for further research into the application of artificial intelligence methods in the field of civil engineering, as it

offers a new method for classifying and predicting the characteristics of construction materials such as concrete. In future studies, the dataset created in this study can also be used for regression analyses.

Keywords: Machine Learning, Deep Learning, Sulfur-based Concrete, Image Classification, Convolutional Neural Networks

ÖZ

Bu çalışmada, betonun mukavemetini, dayanıklılığını ve geçirgenliğini etkileyen önemli bir özellik olan gözenekliliğe dayalı kükürt bazlı beton numunelerinin sınıflandırılması için derin öğrenme ve makine öğrenme tekniklerinin uygulanması araştırılmaktadır.

Araştırmanın ilk kısmında, kükürt bazlı beton numunelerden oluşan bir veri tabanı oluşturmaya ve gözeneklilik gibi özellikleri hesaplamaya odaklandı. Daha sonra gözeneklilik yüzdesi hesaplandı ve görüntüler, gözeneklilik yüzdesine göre düşük gözeneklilik veya yüksek gözeneklilik olarak etiketlendi. Fiziksel numunelerin görüntüleri, bir veri seti oluşturmak için görüntü işleme teknikleriyle otomatik olarak tanımlandı.

Çalışmanın ikinci kısmında, gözenekliliğe dayalı örnekleri tahmin etmek ve sınıflandırmak için bir sinir ağını eğiterek test etmek amaçlandı. Temel bir Konvolüsyonel Sinir Ağı (CNN) ve AlexNet gibi önceden eğitilmiş bir model kullanılarak somut görüntüler düşük ve yüksek gözeneklilik olarak iki ayrık sınıfına ayrıştırldı. Gözeneklilik, hava boşluklarının ve agregaların beton numunesi boyunca dağılımı olarak hesaplandı. En iyi iki modelin karşılaştırılması ve ağların doğruluk ve diğer performans ölçümlerinin bulunması karışıklık matrisleri kullanılarak yapılmıştır.

Bu çalışmanın sonucu, AlexNet gibi transfer öğrenmeli önceden eğitilmiş modellerin, beton üretiminin daha hızlı ve daha verimli kalite kontrolüne yol açabilecek gözenekliliğe dayalı kükürt bazlı beton numunelerini doğru ve otomatik olarak sınıflandırmak için kullanılabileceğini göstermektedir. Bu çalışma aynı zamanda,

beton gibi yapı malzemelerinin özelliklerinin sınıflandırılması ve tahmin edilmesi için yeni bir yöntem sunduğundan, inşaat mühendisliği alanında yapay zeka yöntemlerinin uygulanmasına yönelik daha fazla araştırma için zemin oluşturmaktadır. Gelecekteki çalışmalarda, bu çalışmada oluşturulan veri seti regresyon analizleri için de kullanılabilecektir.

Anahtar Kelimeler: Makine Öğrenimi, Derin Öğrenme, Kükürt Bazlı Beton, Görüntü Sınıflandırma, Konvolüsyonel Sinir Ağları

... To My Devoted Family

ACKNOWLEDGMENT

I would like to express my heartfelt gratitude to my supervisor, Prof. Dr. Ahmet Rizaner, for his unwavering support, guidance, and encouragement throughout my master journey. His expertise, insights, and dedication have been instrumental in shaping my research and enabling me to achieve this milestone.

Finally, I would like to thank my parents and my devoted friend, Dr. Behnam Rafie, for their unwavering support, encouragement, and love throughout my academic journey. Their constant support and belief in me have been the foundation of my research outcomes.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
CNN	Convolutional Neural Networks
DL	Deep Learning
FN	False Negative
FP	False Positive
HDPE	High-Density Polyethylene
LLDPE	Linear Low-Density Polyethylene
ML	Machine Learning
ReLU	Rectified Linear Units
TN	True Negative
TP	True Positive

Chapter 1

INTRODUCTION

In civil engineering and concrete production, air voids inside the product can be very important for the strength and usability of concrete. To find out these metrics, the first step is to calculate the porosity and find the distribution of the voids and aggregates throughout the different parts of the product. Some studies have been done to detect voids and aggregates to calculate the porosity and find other properties such as compressive strength [1].

Inspection of concrete health and construction is one of the most important aspects of civil engineering. These days, with the help of Machine Learning (ML) and artificial intelligence, inspectors can manage to visit more cases with accurate information about cracks and voids inside buildings or concrete in a shorter time. In the civil engineering field, experts are trying to come up with different combinations of concrete for better strength. One of the most crucial factors in concrete compressive strength is porosity, which can be calculated by experts by counting the number of voids and adding the sum of their diameters.

When we have a large amount of data, this porosity analyses process can be time-consuming. Therefore, we can use deep learning and artificial intelligence to automatically find voids from images and calculate the porosity for each sample. After that, we can train a network with these data to predict the porosity distribution based

on different mixture possibilities and the distribution of aggregate materials inside those samples.

Deep Learning (DL) techniques, specifically Convolutional Neural Networks (CNNs), have been used in research to detect voids and cracks in regular concrete samples [2]. In these methods, images of concrete samples are used as input to train a CNN model to classify the samples as either having air voids or air bubbles inside them. There are also many pre-trained models, such as AlexNet [3] and ResNet50 [4], that can help the learning process and achieve better results in a shorter training time. The use of pre-trained models such as AlexNet or ResNet50 could be very useful in our case, as it can help to speed up the training process.

AlexNet is a type of CNN for image classification. It uses images with dimensions of 227 by 227 pixels and contains eight layers. The first five layers are convolutional layers, and the remaining three layers are fully connected layers. The last output of the network is connected to a 1000-way softmax layer to distribute the results into 1000 class labels. To speed up the training process, a non-linear function called ReLU (Rectified Linear Units) is applied to the outputs of all layers (both convolutional layers and fully connected layers). The other advantages of this network are applying Local Response Normalization (LRN) to help generalization by implementing a form of lateral inhibition and using overlapping pooling through the network to reduce the top-1 and top-5 error rates and prevent the model from overfitting. This model also uses dropout in the first two fully connected layers to help the model learn more features.

Data augmentation is one of the most important aspects of the AlexNet model. The model uses data augmentation techniques such as translating the image, horizontally reflecting the images, and changing the intensity of the RGB channels to reduce overfitting and improve the model's ability to generalize. The model is trained using Stochastic Gradient Descent (SGD) with a stack size of 128 samples, momentum of 0.9, and weight decay of 0.0005. The learning rate is adjusted manually throughout training, typically dividing by 10 when the validation error rate stops improving. The weights in each layer are initialized from a Gaussian distribution with zero mean and a standard deviation of 0.01. Neuron biases in certain layers are initialized with a constant of 1 to provide the ReLUs with positive inputs and accelerate the early stages of learning. Neuron biases in the remaining layers are initialized with the constant 0.

In the Civil Engineering department's material laboratory, samples are stored without any containers and the temperature is not controlled. We can investigate the potential effects of these conditions on our samples, such as the relationship between temperature or humidity and compressive strength of concrete. These samples have been created within a year and they have different built dates. For future studies, new samples can be created with the same mixture by also adding the ages of samples to the dataset. This allows us to determine if the age of the sample affects our program for finding porosity and compressive strength.

The research questions addressed in this thesis are as follows:

- How can we automatically detect the number and diameter of voids in concrete samples to find out the distribution of porosity (low or high)?
- How can we create a classification dataset based on the distribution of voids or porosity?
- Which type of models are most suitable for this classification problem?

In response to the above research question, this thesis makes three main contributions. First, it proposes a novel method for automatically detecting the number and diameter of voids in concrete samples, thereby determining the level of porosity. Second, it creates a classification dataset based on the distribution of voids or porosity. Third, it evaluates the performance of different models for this classification problem.

The other part of our study is to create a usable dataset in the future. Every image processing dataset needs to be labeled for interesting features. The process of labeling is called image annotating. There are many ways and tools for annotating pictures such as text annotation, shapes annotation, brush/pen/pencil tools, highlighter, color masking, stamp, sticky notes/comment boxes, underline/strikethrough, polygon/RoI (Region of Interest), and hotspots. However, the process of labeling images dataset for image processing tasks and using it with machine learning techniques is particularly important and it needs to be more detailed. Therefore, some of the mentioned tools may not be useful for these types of tasks. Bounding Box Annotation is the most used annotation tool, and it is mostly applicable to object detection tasks. It draws a rectangular box boundary around the objects of interest in an image. Another method is Polygon Annotation, which uses polygon shapes around the objects of interest instead of a rectangular box. Semantic Segmentation and Instance segmentation are

also methods for classification and detection tasks. Mask Annotation is another method that uses distinct colors for different classes inside an image. We employed python and other tools to create masked images for this dataset for future studies.

The use of the OpenCV library in Python has been proven to be effective in automatically detecting voids. Identifying the features that will provide the best results is also an important part of this study. This library is also useful for generating dataset and determining diameter counts. It enables the creation of masked images featuring annotated voids. Utilizing the capabilities of this Python library, porosity percentage can be computed through void pixel quantification. Furthermore, it facilitates image preservation and masking for later applications. Civil experts also furnish information about sample composition and porosity. This information can be added to our new dataset for future use via machine learning. With the help of programming languages such as Python and MATLAB, we can convert each image to a binary array and try to classify the porosity of these concrete samples.

The cracks and aggregates within the specimens will not be investigated in this study. Our colleague has conducted various tests, including assessing the impact of water pressure on each sample and other pertinent analyses, within their expertise. As a result, potential future investigations could integrate these tests with a machine learning model to forecast their influence on distinct material compositions. It is worth noting that we have retained two distinct mask images, augmenting our dataset to potentially enhance or diversify our analytical capabilities for various tasks.

The limitation of this idea is the discrepancy between the way machines work and the real world. There are many areas where machine learning can be applied, but it is

challenging to apply it directly to civil engineering problems. Machine learning models that are simulated in the lab often fail in real-world tests. This is usually due to a data mismatch between the data used to train and test the machine learning model and the data it encounters in the real world, a phenomenon known as data shift.

This research can help students and researchers in the material field to simplify the process of detecting patterns in their materials, especially when they are working with large amounts of data. As we mentioned earlier, it can be applied to any engineering discipline to detect patterns and make future predictions. This experimental study aims to shed light on the potential models and methods that can be used for concrete image analysis by creating a state-of-the-art dataset of over 3,000 images. The purpose of this thesis is to first create a unique dataset from concrete samples and then train a network to predict whether each image is of low or high porosity.

This thesis makes significant contributions to the larger body of knowledge in several important ways, including the following:

- **The creation of a dataset:** Here, we present a one-of-a-kind dataset consisting of sulfur-based concrete samples. We place an emphasis on characteristics such as porosity, and created separated mask files which will serve as a valuable resource for future research.
- **Model Benchmarking:** Through a comparison of a basic CNN and transfer learning model, this investigation attempts to provide significant insights into the ideal methodologies for similar classification tasks.

1.1 Thesis Overview

This thesis investigates the results of two different deep learning-based machine learning models for predicting porosity classes of sulfur concrete sample images, namely low porosity, and high porosity. This brief overview section provides an outline of the various chapters and their individual objectives. Chapter 1 introduces the thesis, states the problem, discusses the limitations and benefits of this research, and presents potential future studies. It also provides an overall view of the whole thesis. Chapter 2 offers a thorough literature review of the techniques developed to calculate the porosity inside any type of concrete, especially sulfur concretes. Additionally, this chapter examines previous studies on machine learning models, edge detection techniques, and previous usage of transfer learning algorithms, particularly AlexNet. Chapter 3 outlines the research goals, the novelty of the dataset, and provides an overview of the methodology employed. Additionally, a detailed explanation of methodology used through the study for both creating the dataset and training the network is presented. Chapter 4 presents the separate results for both networks' training and testing phases. It also presents a validation test with 100 excluded images and provides comparison of the models, their results, and metrics. Finally, this chapter discusses the future study potentials related to this type of research and dataset. Chapter 5 concludes this research and discusses the applicability of the dataset and method in real-world and other engineering fields.

Chapter 2

LITERATURE REVIEW

The application of detecting objects through pictures with the help of ML and deep neural networks has been around for more than a decade [5]. It has a very flexible usage from detecting lunar crates inside a telescope picture [6] to identifying atoms inside a microscopic image [7]. However, the usage of ML algorithms for Structural Health Monitoring tasks, such as bridge crack detection and tunnels health conditions did not become popular until 2006 [8].

Images of the surface can be useful for finding the condition of a concrete structure, but there are many other factors that cannot be seen in images. In recent years, experts have developed many techniques to understand what is happening inside the concrete structure. As stated in [9], the best way to obtain the samples from inside is the non-destructive techniques. According to Flah's review article, techniques that are used for condition assessments inside a concrete structure have been divided into two main categories: vibration-based and image based [8]. Vibration-based and image-based techniques can be used for concrete condition assessment. However, as the thesis study is based on detecting voids inside microscopic images of sulfur-based concrete samples, we focused on Imaged based techniques.

Until now, some studies have investigated cracks inside concrete sample images based on their features, these features can be extracted for any images and also, they are

different in many criteria, such as patterns [11] [12], using multi-resolution analysis to select cracks inside images [13] or finding edges to help identify diameter of cracks inside a sample for detecting the patterns [14]. However, in 2021, an automated crack pattern recognition method based on DISTS index (a CNN method) was invented to classify features of cracks based on structural or non-structural patterns inside images from real world and it can predict cracks with 96% accuracy [11]. One of the biggest challenges is to automatically identify cracks from an image containing actual cracks and crack-like noise patterns such as dark shadows, stains, lumps, and holes, which are often seen in concrete structures. In Kim's study, authors came up with a method that filters input images based on the Hessian matrix to remove noises and the outcomes showed it can easily classify crack from non-cracks based on the shape and direction of them [14]. For this study, we have access to more than a hundred samples of a new type of concrete based on sulfur. Our aim in this study is to detect air voids inside them to calculate the porosity. Based on the porosity and the mixture of samples, the compressive strength of the future samples also can be predicted [15].

A comparison for prediction of compressive strength of concrete sample has been carried out in [15] using ML techniques via "R" software environment and employed three-dimensional reconstruction methods for mapping the images to machine.

Active and passive are the main subcategories for 3D reconstruction methods [16]. The first researcher who studied the automated version of this type of photography was Wolter [17]. However, one of the most popular ways for obtaining pictures, which is Binocular Stereo, was invented by Biskup [18]. Since we want to obtain our images with a microscope or smartphone, this method is well suited for such applications. In another study, Girshick [19] presented the images as pixel segmentation to detect

objects accurately by hierarchy of features. This algorithm has since been used in many areas such as autonomous driving [20], face detection [10], and image search engines [21].

The main advantage of this method with ML and especially Deep Learning (DL) is the ability to learn the features of each pixel and use that to detect objects inside the pictures more accurately [19]. In contrast to reconstruction methods, there are other methods that can detect objects with other features such as Crater Detection Algorithm (CDA) for detecting lunar craters. Although this algorithm has a completely different nature with our concrete samples, it can still be useful for our research to detect voids inside the images. As we mentioned before, the CDA algorithm is a crater detection algorithm that can find the lunar crater and based on the diameter and other features it can prepare a catalogue about that [22]. This algorithm can be used in our study for finding the diameter of voids. In the other area, Yaman studied the calculation of porosity based on the voids inside the concrete samples and for the accurate result they need to find more than 2,000 voids and their diameter inside each sample, which makes this task a time-consuming and labor-intensive process [23].

Although many features can be used, finding compressive strength is one of the most prominent features for concrete production and predicting compressive strength based on porosity and mixture. These techniques to detect voids inside concrete samples have been studied in [15] and are compared together based on accuracy and speed. The method called Image thresholding in OTSU algorithm changes the image to black and white and defines a threshold value to separate the voids from the rest of the image [24].

2.1 Edge Detection

Edge detection algorithms are used to detect the edges of the voids inside an image. There are some popular techniques such as Canny edge detection or Sobel edge detection [13]. Another algorithm called Blob detection is also useful for identifying the voids in the image. These algorithms look for connected areas based on the pixels that have similarity in the color or intensity and can be used to locate the voids in the image [5].

2.2 Transfer Learning

In the past decade, CNNs have revolutionized the field of image classification. The hierarchical structure of CNNs allows them to learn spatial hierarchies of features automatically and adaptively from images [3]. Krizhevsky's work on the AlexNet was instrumental in propelling the utility of deep learning in image classification tasks.

The scholarly work written by Krizhevsky [3], commonly referred to as the AlexNet paper, was a notable milestone in the domain of deep learning and computer vision [3]. Before 2012, the dominant approaches used for addressing image classification tasks used conventional image processing methods and shallow machine learning algorithms.

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) was a popular benchmark where these methods were tested. The primary contribution of this paper was the introduction of a deep CNN called AlexNet, which significantly outperformed all the other techniques in the 2012 ImageNet challenge. The network architecture was deep (compared to prior standards) with 8 layers - 5 convolutional layers followed by 3 fully connected layers. This study used the Rectified Linear Unit (ReLU) as the

activation function, which was found to train the network several times faster than tanh and sigmoid. To reduce overfitting in the fully connected layers, they introduced the "dropout" technique, a regularization method where a subset of neurons is randomly ignored during training, which helps in achieving a more robust model. Additionally, the team used data augmentation techniques like random cropping, flipping, and RGB color shifts to artificially expand the training dataset, improving performance and robustness.

Localized response normalization helped the model handle a wide range of activations, ensuring that the neuron outputs don't become too high or too low. Instead of the common practice of using average pooling, the team used overlapping max pooling, which reduced the top-1 and top-5 error rates.

Due to the model's deep architecture, traditional CPUs were insufficient for training. Authors in this study used two Nvidia GTX 580 GPUs for training, marking one of the pioneering uses of GPUs for deep learning tasks. The design was such that the GPUs communicated only in certain layers, ensuring efficient parallel computation. The massive reduction in error rate achieved by AlexNet on the ImageNet challenge was groundbreaking. The success of this deep learning approach spurred a flurry of interest and subsequent advancements in deep learning, solidifying neural networks' position as the go-to technique for computer vision problems.

Transfer learning is very useful for many applications like object recognition [26], identifying logos inside an image [27], and descriptive text generator for an image [28]. For training a network faster and with less data than a common neural network there are pre-trained models such as resNet50 or AlexNet. The study of these kinds of

networks has been done previously and showed that these networks can be useful for this goal in training a network.

2.3 Convolutional Neural Networks (CNNs) for Image Classification

Image classification has been a central topic of research in computer vision for several years. The task involves categorizing a given image into one of several predefined classes. The classification can be binary (where the image can belong to one of two possible categories) or multi-class (where the image can belong to one of more than two categories). In this part of the literature review, we discuss some of the pivotal works that have shaped the development of image classification for both binary and multiple classes.

The process of binary image classification, which involves the classification of images into two separate classes, is commonly utilized as an initial stage in many classification tasks. Image classification has diverse applications across several fields. For instance, it is utilized in medical imaging to differentiate between abnormal and normal tissues. Additionally, it is employed in spam detection to identify photos as either spam or non-spam. The publication by Razavian [29] is widely regarded as a foundational contribution to the domain of deep learning for binary categorization. The research demonstrates the effectiveness of employing characteristics derived from pre-trained convolutional neural network (CNN) models for various image recognition tasks.

The incorporation of many classes into binary classifiers poses intrinsic challenges. One often adopted approach entails the utilization of the one-vs-all strategy, whereby a binary classifier is trained for each specific class in contrast to all remaining classes. However, it is important to acknowledge that CNNs have exhibited their proficiency

in efficiently managing numerous categories. The phenomenon is demonstrated through the notable accomplishments of prominent networks such as VGG, Inception, and ResNet in the ImageNet challenge, which encompasses an impressive array of 1000 classes [30], [31], [32].

With the rapid progress in deep learning, there are emerging trends and challenges in image classification. Issues like interpretability of models, adversarial attacks, and incorporation of attention mechanisms have gained traction [33].

The paper "Attention is All You Need" by Vaswani [34] introduced a novel architecture called the "Transformer" which has since become the backbone for many state-of-the-art models in Natural Language Processing (NLP) and even some image-related tasks. Before this work, most sequence-to-sequence tasks in NLP were dominated by recurrent models like LSTMs or GRUs. While effective, these models process sequences step-by-step, which can be computationally intensive, especially for long sequences. The Transformer architecture replaced recurrence with the self-attention mechanism. In layman's terms, instead of processing sequences word-by-word, the self-attention mechanism enables the model to concentrate on different parts of the input text at the same time. This "attention" distributes importance to various input elements based on their relevance, enabling more parallel processing, and capturing dependencies regardless of their distance in the sequence. The architecture itself consists of an encoder and decoder stack. Each encoder has multiple identical layers with two sub-layers: a multi-head self-attention mechanism and a simple position-wise fully connected feed-forward network. The decoder has an additional multi-head attention layer to attend to the encoder's output. Importantly, to maintain sequence order without recurrence, they introduced position encoding, added to the input

embeddings at the bottoms of the encoder and decoder stacks. This work led to significant improvements in translation tasks, setting new state-of-the-art benchmarks. Its architecture's scalability and effectiveness soon became foundational, leading to models like BERT, GPT, and many others, revolutionizing the NLP field. While the paper primarily dealt with NLP tasks, the concept of attention and the Transformer architecture found its way into other domains, including computer vision. For image classification and generation tasks, certain modifications and adaptations of the Transformer model have been explored. In essence, [34] shifted the paradigm by demonstrating the power of attention mechanisms, proving that models don't necessarily need to rely on recurrence or convolution to achieve high performance in sequence-based tasks.

Image classification, whether binary or multi-class, has witnessed significant advancements with the advent of deep learning. While CNNs have dominated the landscape, continuous research has led to refinement in techniques, promising even better performance and more generalized applications in the future.

Chapter 3

METHODOLOGY

This chapter explains the methodology employed in the thesis. It begins by detailing the sample creation process. Subsequently, the chapter delves into the procedures involved in capturing and preparing a dataset of over 3,600 JPEG-formatted images. The annotation of these images is then discussed, with an exploration of both automated and manual annotation methods, as well as the creation of masks for future image segmentation studies. This chapter also outlines the calculation of porosity based on the distribution of air voids within the sulfur-based concrete samples. Lastly, the process of labeling the dataset as either low or high, with the assistance of a civil expert, is described. The upcoming session at the end will focus on the process of testing and training using two specific models, namely the Convolutional Neural Network (CNN) and AlexNet. Additionally, the session will cover the creation of an additional independent test set to enhance the accuracy of the outcomes. Lastly, the session will discuss the rationale of utilizing CNNs and Transfer Learning.

3.1 Samples Creation and Preparation Process

Samples of this study are created in the laboratory of Civil Engineering department of Eastern Mediterranean University laboratory from sulfur-based concretes. These types of concrete are created by combining different aggregates, natural materials such as gravel, crushed stone, and recycled concrete and chemical materials such as High-Density Polyethylene (HDPE) polymer or Linear Low-Density Polyethylene (LLDPE) polymer inside a temperature-controlled heating mixer as shown in Figure 3.1.

Figure3.1: Temperature-controlled Heating Mixer

These aggregates help to increase the strength of the concrete products by providing a framework for the hardened material, as well as helping to reduce shrinkage and cracking and the porosity is the air voids between these aggregates that will reduce the overall strength of the products. The created samples (see Figure 3.2) were sliced into 5 cm sections for the next step. The next stage before the image capturing part was washing the sliced samples surface with water and drying them out. This helps remove any extra dust or aggregates remained after slicing process and increases the visibility of voids and aggregate's colors on the surface of concrete samples.

Figure3.2: Sulfur-based Concrete Samples

3.2 Dataset Creation and Preparation Process

The smartphone camera was utilized in Macro mode, positioned on a magnetic stand, and operated using voice command functionality. The camera was placed 10 cm away from the sample on a white surface. The magnetic stand and voice command feature are employed to save images without altering the camera's position. This is important because it ensures that the images are captured with a high level of detail. Some examples of images captured using this method are visible in Figure 3.3.

Figure 3.3: A Sample Captured Image in Original Size

The utilization of near zero angle lighting was implemented in the image capture procedure, as per the investigation conducted by Hong. This methodology entailed the utilization of the side lights of a microscope that was equipped with adjustable illumination tubes, as shown in Figure 3.4. The study conducted by Hong emphasized the effectiveness of this lighting technique in capturing photographs where voids have a deeper appearance as a result of shadow presence. The objective of utilizing this methodology was to streamline the process of automatic annotation and improve the overall efficiency of the OpenCV library [25].

Figure 3.4: Microscope with Adjustable Lights

The dimensions of the original captured images are 1160 by 2576 pixels. This image size is quite sufficient to capture the voids with other details such as distribution of aggregates from surface of the concrete. A total of 1800 images in jpg format were obtained from over 100 samples. The images were captured from various sections of the samples, covering both low and high porosity regions. Additionally, different areas of the samples were included based on the color and distribution of aggregates. This approach aimed to enhance the diversity of the dataset and facilitate the extraction of additional features by the model, as depicted in Figure 3.5.

Figure 3.5: Different Parts of The Samples

Subsequently, the photos were divided in half to extend our dataset and optimize the images for the training phase. A sample cropped image is available in Figure 3.6. This process enabled us to increase the sample size of our dataset to 3600 images in jpg

format. In the preparation phase, a thorough examination was conducted on all 3600 photographs in order to identify and eliminate those that exhibited noise, blurriness, incompleteness, or were deemed useless. Some examples of these unusable images are shown in Figure 3.7.

Figure 3.6: Cropped Image to Half Size

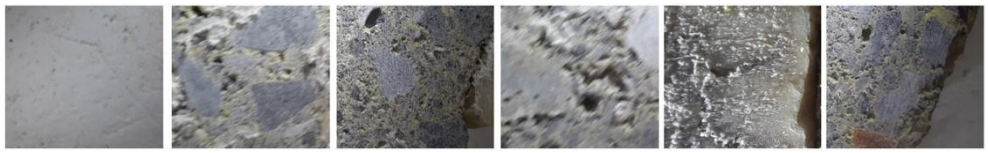

Figure 3.7: Example of Removed Images from Dataset

3.3Image Annotation and Mask Creation

The succeeding applications involved the utilization of the OpenCV library in the Python programming language to find and annotate the air voids, following the image cropping procedure. The first step was to find the voids inside images by converting them to grayscale and trying to fill the voids based on their darker color and shadow. After many different tries, this goal was achieved with the mentioned threshold in Figure 3.8. The detection in this part mostly fulfills the function of automatically annotating the photos.

Figure 3.8: Suitable Threshold for OpenCV2 Library

The diagram is for the code snippet: “ret, thresh = cv2.threshold(gray, 55, 255, cv2.THRESH_BINARY_INV)” of OpenCV2 Python Library. This code applies a thresholding operation to a grayscale image using OpenCV library. The resulting image is stored in the variables "ret" and "thresh". The thresholding is performed using a threshold value of 55, where pixel values below this threshold are set to 255 (white).

As can be seen in Figure 3.9, the automatic detection is not 100% accurate. The mistakes were fixed by manually annotating the images with image editing tools such as Adobe Photoshop. This careful annotation process yielded excellent results, as demonstrated in Figure 3.10.

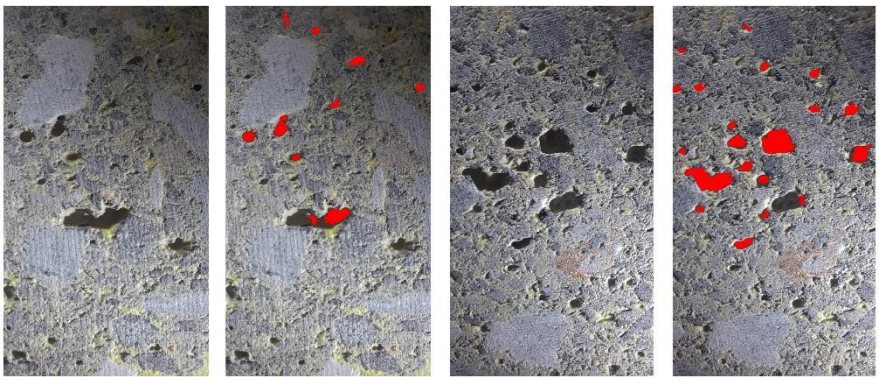
Figure 3.9: First Try to Annotate the Images Automatically

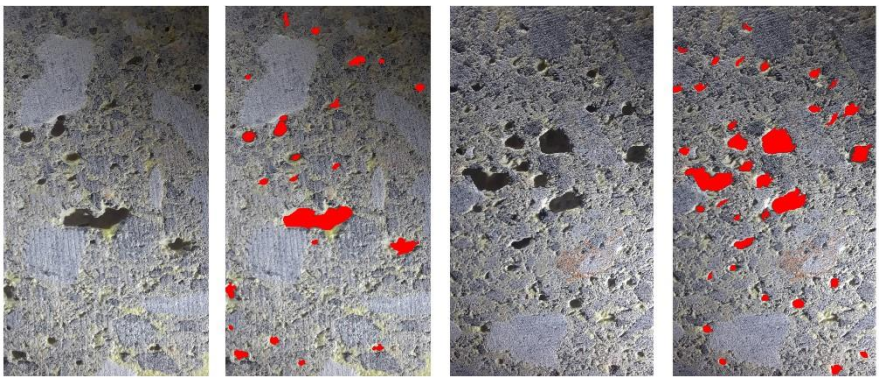

Figure 3.10: Second Try to Annotate the Images Manually

In the process of annotating the images, red pixels were used to represent voids. The same red color was used to manually fill in any undetected voids inside the samples. the mistakenly detected aggregates were also removed to clarify our dataset and calculate percentage of the porosity.

The main purpose of filling the voids was to count the number of red pixels later to calculate porosity percentage of that sample. Furthermore, the annotated images in both monochrome (black and white) and colorized versions (purple and yellow) were saved for dataset further applications. As mentioned before, there are many ways to annotate an image for computer vision tasks. One the most useful methods is to create mask images based on distinct colors. Different masked versions can be seen in Figure3.11 and Figure3.12.

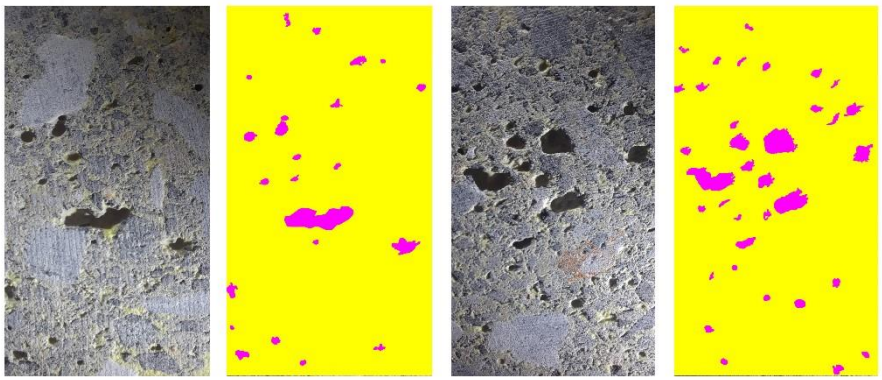

Figure 3.11: Purple and Yellow Masks

Figure 3.12: Black and White Masks

3.4 Porosity Calculation

After the image annotation and mask creation process, another OpenCV Python script was used to calculate the sum of the red areas in pixels and save it with the equivalent image name in an Excel file. The porosity percentage was then easily found by dividing the sum of the red areas by the area of the image. The first few rows of the created dataset are shown in Figure 3.13.

$$porosity\ (\%) = \frac{sum\ of\ red\ pixels}{sum\ of\ all\ pixels} \times 100$$

The concept of porosity in materials such as concrete refers to the presence of empty spaces or voids within the material. In the present study, the aim was to ascertain the percentage of porosity by evaluating the ratio of voids (shown by red pixels in the annotated photos) to the total area of the concrete sample.

The annotated images used in our study used red pixels to indicate voids present in the concrete. Therefore, the sum of these red pixels provides a direct measurement of the total empty area present in the image. The area of an image can be determined by multiplying its width and height in pixels. The area depicted in the illustration encompasses the entirety of the sample area, encompassing both solid and empty spaces.

The relative fraction of voids in the sample can be determined by calculating the ratio of the sum of red pixels (representing the void area) to the total image area. This ratio yields a numerical value ranging from 0, indicating the absence of voids, to 1, representing a complete void presence. To express a proportion as a percentage, it is necessary to multiply the ratio by 100. This conversion allows for a more easily comprehensible measure in the context of porosity analysis.

The justification for using this formula is based on the fundamental concept of percentage calculation. This principle involves dividing the part (voids) by the whole (total sample area) and then multiplying the result by 100 to represent it as a percentage. The formula provides a numerical assessment of porosity, facilitating the ability to make comparisons and conduct further research. In the context of practical applications, this percentage can be likened to the process of physically quantifying the empty spaces within a given sample and then representing them as a proportion of the sample's overall volume.

The consideration of porosity in the design of concrete mixes is a crucial factor that significantly impacts the overall performance and long-term durability of concrete structures. Various techniques exist for determining the porosity of concrete

specimens, which are dependent on factors such as the sample's type and dimensions, the available instrumentation, and the desired level of precision.

The helium pycnometer method is widely employed for determining the porosity of concrete samples. The procedure involves quantifying the volume of a concrete specimen with a helium pycnometer, a specialized apparatus that employs helium gas as a medium. The porosity of the concrete sample (φ) can be calculated using the following formula [35]:

$$\varphi = \frac{Vh - Vb}{Vh}$$

Where the volume of helium gas displaced by the concrete sample is denoted by Vh, and the bulk volume of the concrete sample is denoted by Vb, both in centimeters.

The study used a pixel-based approach to estimate porosity. This provides a computerized method for measuring a parameter that is typically measured through more time-consuming physical measurements. By utilizing image processing techniques and the pixel-based representation of annotated images, it is possible to effectively compute the porosity percentage for a significant quantity of concrete samples.

With the help of a civil engineering expert, all the samples with porosity less than or equal to 3.5 percent were labeled as low porosity class, and the others as high porosity class as shown in Figure 3.13.

	A	B	C	D	E	F	G	H
1	Image Name	Total Area	Sum of Red Areas	porosity	class #	class binary	class label	
2	sample (1).jpg	1494080	20014	1.339553	0	01	low	
3	sample (2).jpg	1494080	238038	15.93208	1	10	high	
4	sample (3).jpg	1494080	83983.5	5.621085	1	10	high	
5	sample (4).jpg	1494080	93493.5	6.257597	1	10	high	
6	sample (5).jpg	1494080	116466.5	7.795198	1	10	high	
7	sample (6).jpg	1494080	124165.5	8.310499	1	10	high	
8	sample (7).jpg	1494080	51715	3.461327	0	01	low	
9	sample (8).jpg	1494080	92756.5	6.208269	1	10	high	
10	sample (9).jpg	1494080	81353.5	5.445056	1	10	high	
11	sample (10).jpg	1494080	80442.5	5.384083	1	10	high	
12	sample (11).jpg	1494080	208134	13.93058	1	10	high	
13	sample (12).jpg	1494080	185430.5	12.41102	1	10	high	

Figure 3.13: First Few Rows of the Dataset

3.5 Training and Testing

Two distinct models were employed in the training process of the network to classify the data according to the given labels. In order to conduct the training and testing procedure, MATLAB was chosen due to its beneficial attributes of speed and simplicity, particularly in the context of classification problems.

In the first attempt, a basic CNN with 2 convolution layers was used as supervised learning with provided labels. RGB coded images of size 227x227 pixels were presented to the network with a learning rate of 0.0001 as a constant value, a batch size of 16, and the network was trained for 100 epochs to calculate accuracy. The initial layer of our CNN consisted of a 2D convolution layer. This layer included 20 filters, each with a kernel size of 5x5 pixels. A batch normalizer layer was then used to normalize the activations of the neurons in the convolutional layer. In the implemented code, after batch normalizer layer the ReLU activation function was employed to the inputs. A 2D Max pooling layer was then used to reduce the spatial dimensions (width and height) of the input volume to 2x2 pixels and contain important features of the input for the next convolution layer. The last part sends the input features to the 50 classes followed by another ReLU activation function. This layer is then connected to a fully connected layer with 2 classes. Finally, the softmax function was used on the

outputs of the previous fully connected layer. The softmax function transforms its inputs into a probability distribution, meaning that the output of each neuron will be a value between 0 and 1, and the sum of all the neuron outputs will be 1. This makes it easy to interpret the output of each neuron in the final fully connected layer as the network's confidence that the input belongs to the respective class. The samples were also shuffled every epoch to get better results during the training process. The layers of this architecture can be seen in Figure 3.14.

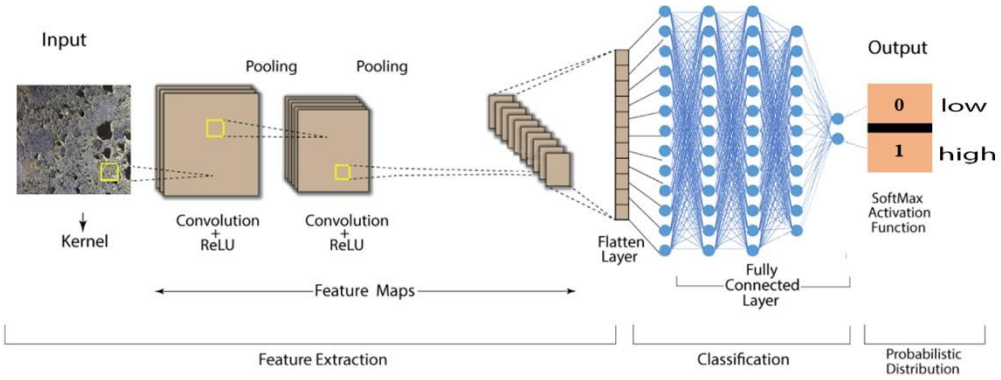

Figure 3.14: CNN Model Architecture

In the subsequent network, transfer learning was implemented by leveraging a pre-trained model known as AlexNet, which is available in the Deep Learning Toolbox of MATLAB. The input images were presented as 227 by 227 pixels in RGB colored format with the same criteria explained before. The learning rate was 0.0001, the batch size was 16 and the network was trained for 100 epochs before reaching the results.

To take advantage of the features that the pre-trained AlexNet model had learnt and use them as a springboard for a new job (a process known as transfer learning), the final three layers of the pre-trained network were used. The final layers of the pre-trained network, which stand in for fully connected layers, were then taken out and

replaced with new layers that are better suited to the brand-new task of identifying porosity.

Following that, a fully connected layer with 2 neurons per class was built to represent low and high porosity. The learning rates, weights and biases in this layer were specially modified using the parameters "WeightLearnRateFactor" and "BiasLearnRateFactor", both of which were set to 20. The global learning rate was multiplied by the learning rate factor. Therefore, in this instance, the weights and biases in this layer would learn at a rate that is 20 times faster than the overall learning rate. For both networks, 80 percent of the images were used for training, 20 percent of the images were used for testing and, and around 100 images were kept out of this process for future testing and validation. The layers of the AlexNet architecture are presented in Figure 3.15.

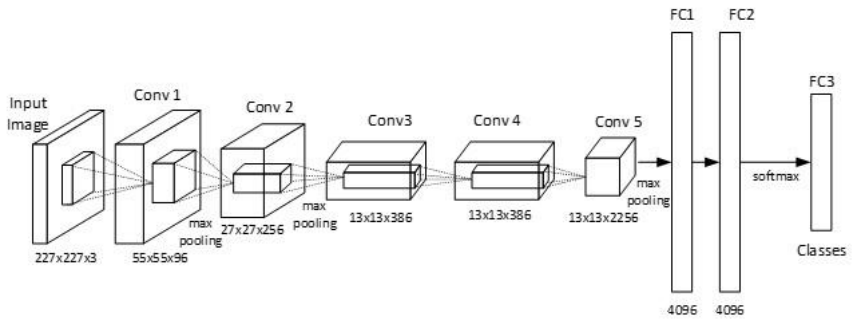

Figure 3.15: AlexNet Model Architecture

3.6 Validation Procedure

A crucial aspect of our investigation was to evaluate the robustness and generalizability of the constructed models. This was achieved through a meticulous validation procedure. Here is a comprehensive explanation of our validation methodology:

- The selection of cross-validation was made to evaluate the performance of our model on data that had not been previously seen. Cross-validation is a widely accepted technique in the field of machine learning. The proposed approach involves dividing the initial dataset into two distinct subsets: an 80% training set, which is used to train the model, and a 20% validation set, which is used to assess the model's performance. By performing this action iteratively and calculating the mean of the results, we have a more precise approximation of the performance of the model.
- The binary classification problem was established with the primary aim of categorizing concrete samples into two distinct groups, namely low and high porosity. The default binary classification setting in MATLAB was used to ensure that the validation procedure adhered to industry norms.
- The validation process was conducted every 120 epochs during the training phase. An epoch is defined as a full iteration in which all training instances are sent through the network in both forward and backward directions. By conducting validation at regular intervals of 120 epochs, we implemented a systematic approach to continuously check the performance of the model. The regular validation process facilitated the early detection of any indications of overfitting and verified that the model was effectively acquiring the desired features from the dataset.

- The selection of AlexNet, a widely recognized neural network design, was based on its documented efficacy in the domain of image categorization tasks. The selected software, MATLAB, provides comprehensive support for AlexNet, facilitating smooth integration and optimizing the training and validation procedures. In addition, we relied on the optimal parameters of MATLAB's default settings, which have been developed through extensive research and practical implementation in various real-world scenarios.
- The outcome of the validation process was assessed by periodically analyzing the validation results, which were generated every 120 epochs. These results served as a diagnostic tool for evaluating the performance of the model. If there is a substantial discrepancy between the model's performance on the validation set and its performance on the training set, this could suggest the presence of difficulties such as overfitting. Consequently, it may be required to make improvements to the model.

In brief, the validation procedure was meticulously constructed to ensure comprehensiveness and rigor, utilizing the advanced functionalities of MATLAB and the well-established architecture of AlexNet. By employing diligent monitoring and employing cross-validation techniques, we ensured that the predictions generated by our model were not only precise when applied to the training data, but also generalizable to novel, unobserved data.

Chapter 4

RESULTS and DISCUSSIONS

The training procedure involved the utilization of two distinct networks: a simple CNN and a pre-trained model known as AlexNet. Both networks were trained and assessed on a dataset of images representing low and high porosity sulfur-based concrete. The networks were subjected to further examination by utilizing an independent set of images in order to calculate the metrics. The results obtained from each model are presented and discussed in the following sections.

4.1 Customized Simple Convolutional Network

The Convolutional Neural Network model explained in Chapter 3 was trained over 80, 100, and 160 epochs with different criteria, such as different batch size and different learning rates. The best results were achieved with 100 epochs, a constant learning rate of 0.0001 and a batch size of 16. Over this period, the accuracy of the model gradually improved, as visualized in the training accuracy chart (see Figure 4.1). By the end of the training process, the CNN achieved an accuracy rate of 85.86% on the test data, demonstrating a respectable level of performance in classifying concrete samples based on porosity. The blue line depicts the accuracy rate observed during the training phase, while the black line represents the validation accuracy recorded at intervals of 120 epochs.

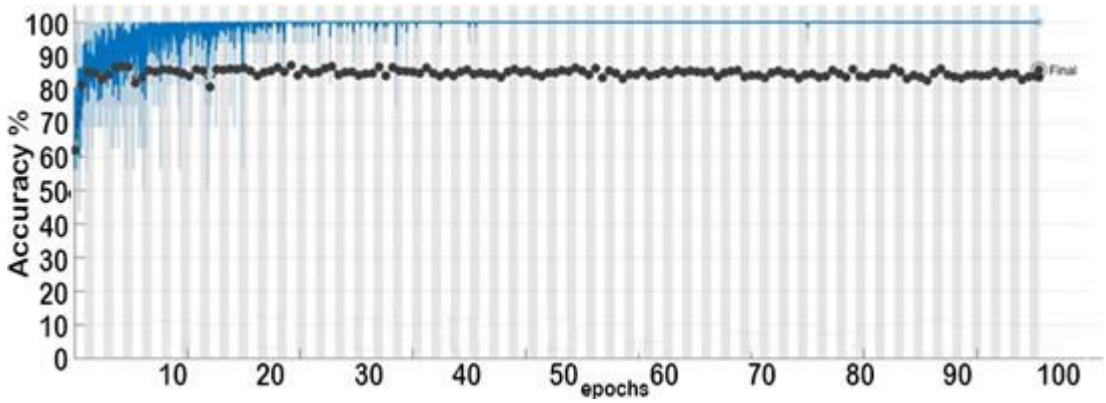

Figure 4.1: Training Accuracy of Proposed CNN Model for 100 Epochs

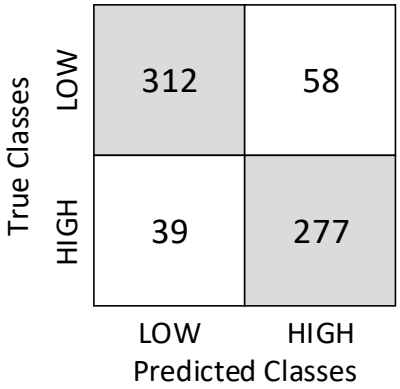

Figure 4.2: CNN Model Confusion Matrix (After 100 Epochs)

The confusion matrix (see Figure 4.2) highlights some instances of misclassification.

4.2 AlexNet Model

The AlexNet model was trained with the same training parameters as the CNN model, including the constant learning rate of 0.0001, a batch size of 16 and 100 epochs. As shown in the training accuracy chart (see Figure 4.3), the AlexNet model achieved a higher accuracy rate of 93.15% on the test data than the CNN model.

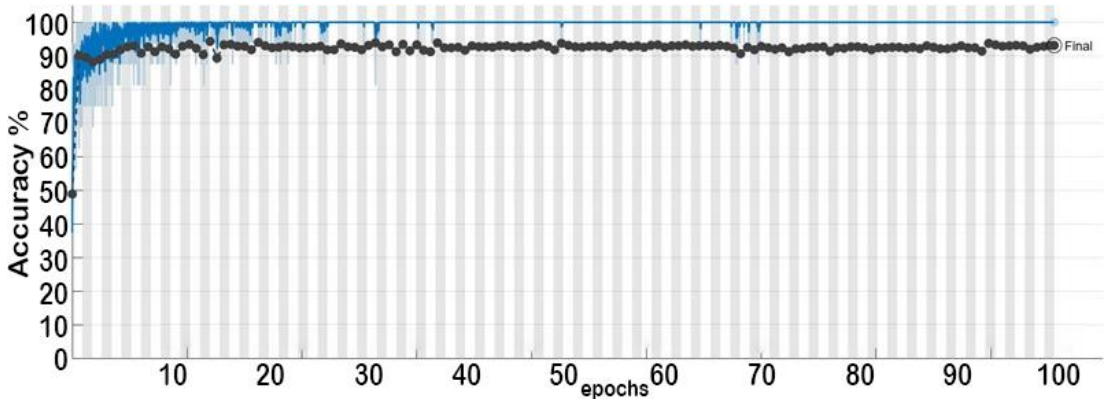

Figure 4.3: Training Accuracy of Proposed AlexNet Model for 100 Epochs

The confusion matrix for the AlexNet model (see Figure 4.4) indicates fewer instances of misclassification compared to the CNN model, confirming its superior performance on this task.

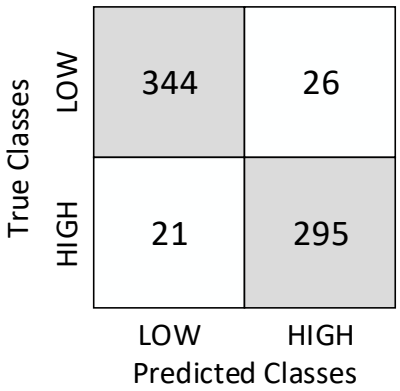

Figure 4.4: AlexNet Model Confusion Matrix (After 100 Epochs)

4.3 Performance Metrics

Based on the parameters provided in confusion matrix shown in Figure 4.5, it is possible to compute five performance metrics, namely Accuracy, Precision, Recall, F1-Score, and Specificity.

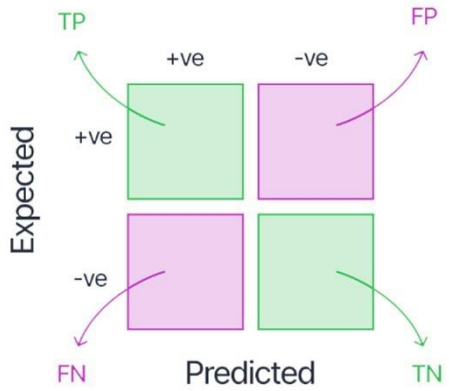

Figure 4.5: Binary Class Confusion Matrix Schema

True Positive (TP): samples had been classified correctly for positive class.

False Positive (FP): samples had been classified correctly for negative class.

True Negative (TN): samples had been classified falsely for positive class, but they belong to negative class.

False Negative (FN): samples had been classified falsely for negative class, but they belong to positive class.

Accuracy: this metric tells us how many samples out of all the ones in the test set were correctly classified.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

Precision: This metric tells us how many samples that the model predicted to be in the positive class actually fell into that category.

$$\text{Precision} = \frac{TP}{(TP + FP)}$$

Recall: This metric tells us the proportion of samples really belong to the positive class out of all samples. In other words, it tells us the percentage of samples that were correctly classified as positive.

$$\text{Recall} = \frac{TP}{(TP + FN)}$$

F1-Score: This metric gives us the sum of the accuracy and recall scores for the class that performed positively.

$$\text{F1} - \text{Score} = \frac{(2 \times Precision \times Recall)}{(Precision + Recall)}$$

Specificity: This metric tells us the proportion of samples in the data set that were correctly predicted to be in the negative class, out of all samples in the data set.

$$Specificity = \frac{TN}{(FP + TN)}$$

The metrics were derived based on the confusion matrices of both networks.

4.4 Performances of Deep Learning Models

The results for the CNN model are depicted in Figure 4.6. The outcomes for the AlexNet model can be seen in Figure 4.7. Based on the results in Table 4.1, the comparison chart illustrated in Figure 4.8, presents a comparative analysis of the two networks conducted over a span of 100 epochs. It is evident that the AlexNet model showed superior performance compared to the CNN model across all metrics. The pre-trained AlexNet model's higher accuracy is due to its existing knowledge base, which allowed for a more effective transfer learning process.

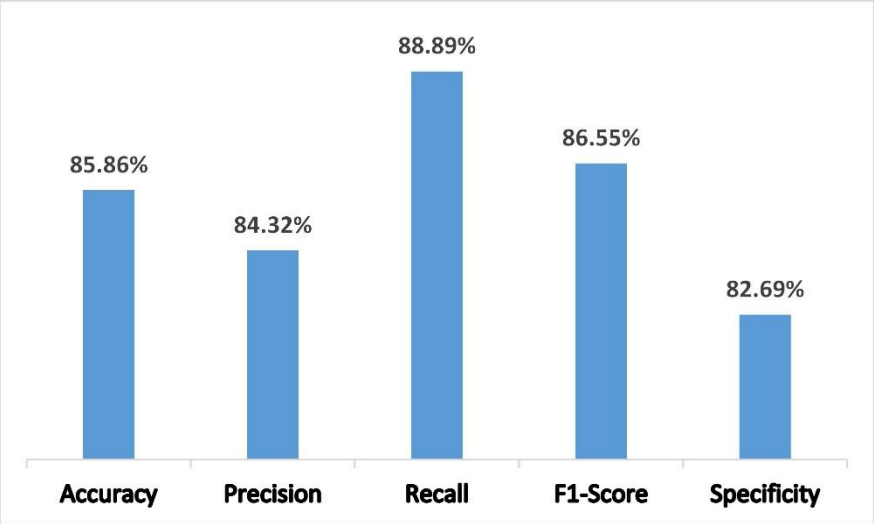

Figure 4.6: CNN Model Metrics After 100 epochs

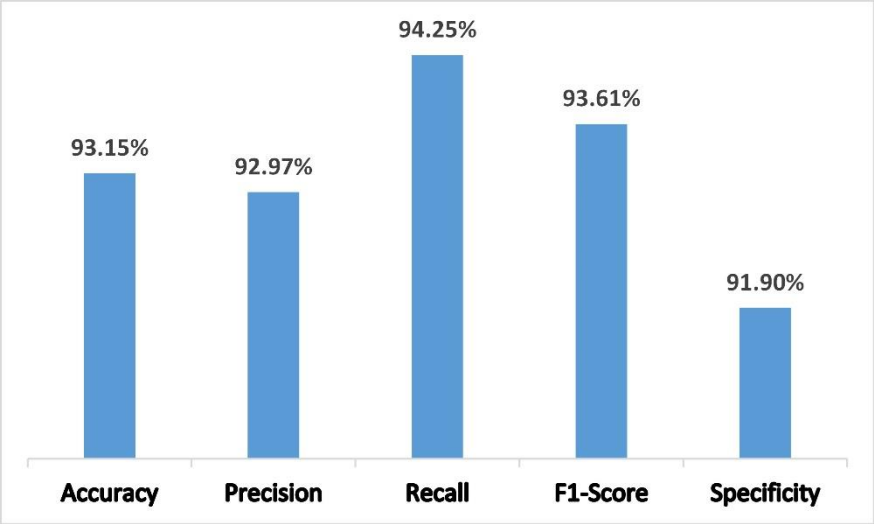

Figure 4.7: AlexNet Model Metrics After 100 epochs

Table 4.1: Results for 100 Epochs

	AlexNet	**CNN**
Accuracy	93.15%	85.86%
Precision	92.97%	84.32%
Recall	94.25%	88.89%
F1-Score	93.61%	86.55%
Specificity	91.90%	82.69%

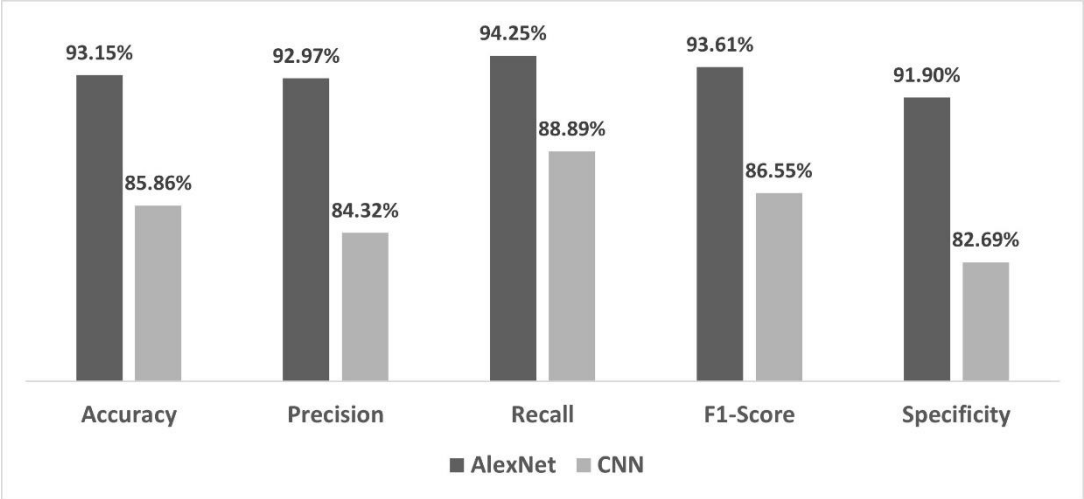

Figure 4.8: Performance Comparison of Network Models After 100 Epochs

Comparing the performance of the two models, AlexNet clearly outperformed the employed CNN model, demonstrating a higher classification accuracy of 93.15% against 85.86%. The superior performance of AlexNet can be attributed to its pre-trained network's extensive learning on a large-scale image dataset. This gave it a distinct advantage, especially in the feature extraction stage of the model.

The loss charts for both models (see Figures 4.9 and 4.10) visually represent the progression of loss values throughout epochs during the training process, with the orange line serving as a depiction of this evolution. The loss function measures the discrepancy between the predicted outputs produced by the model and the actual target values found in the training dataset.

The primary objective during the training procedure of a neural network is to minimize this loss. As time progresses, there is a noticeable pattern of diminishing loss, indicating that the model is gaining knowledge and improving its ability to make predictions. It also offers insights into the learning process by providing an understanding of how quickly or slowly the models learned to classify the images and

where potential adjustments to the learning rate or other parameters might be beneficial.

The loss value of the model on the validation dataset is represented by the black line in the loss plots, which visually demonstrates the evolution of training over epochs. The validation dataset is a separate and unique set of data that has not been utilized by the model throughout the training process. The objective of employing this methodology is to evaluate the efficacy of the model on unseen data, while also monitoring the presence of overfitting.

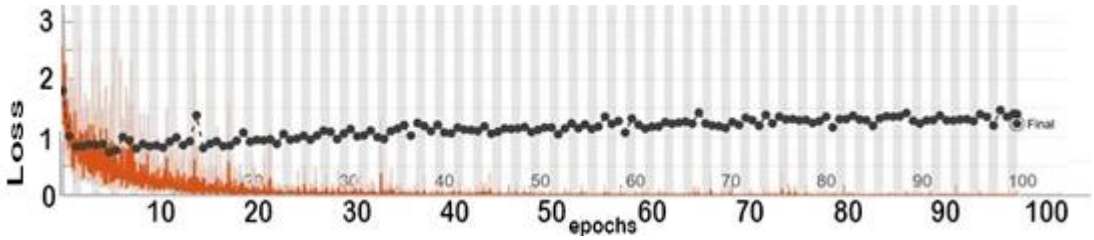

Figure 4.9: CNN Model Training Loss

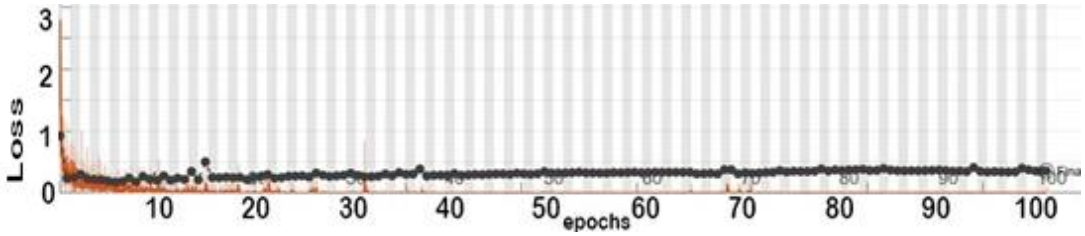

Figure 4.10: AlexNet Model Training Loss

The effect of varying the number of epochs on the training process has been studied. We evaluated the performances of the networks at 80, 100, and 160 epochs with similar hyperparameters. The results for the first training with 160 epochs and its hyper parameters are shown in Table 4.2.

Table 4.2: Performance Evaluation of The Models After 160 epochs (Learning rate=0.0001, Batch size=16)

	AlexNet	**CNN**
Accuracy	91.69%	82.80%
Precision	91.89%	79.19%
Recall	92.64%	87.72%
F1-Score	92.27%	83.24%
Specificity	90.60%	78.13%

Based on the data shown in Table 4.2, it is evident that the performance of our network in terms of porosity classification is superior when trained for 100 epochs compared to a higher number of epochs.

The results for the second training with 80 epochs and its hyper parameters are shown in Table 4.3.

Table 4.3 Performance Evaluation of The Models After 80 epochs (Learning rate=0.0001, Batch size=16)

	AlexNet	**CNN**
Accuracy	92.42%	85.13%
Precision	89.19%	80.81%
Recall	96.49%	90.61%
F1-Score	92.70%	85.43%
Specificity	88.37%	80.06%

As previously seen in Table 4.1, the results obtained with 100 epochs are slightly better than the results obtained with 80 epochs. Therefore, the optimal number of epochs for training was selected as 100 for classify porosity.

The comparison of confusion matrices for different number of epochs is presented in Figure 4.11 and 4.12.

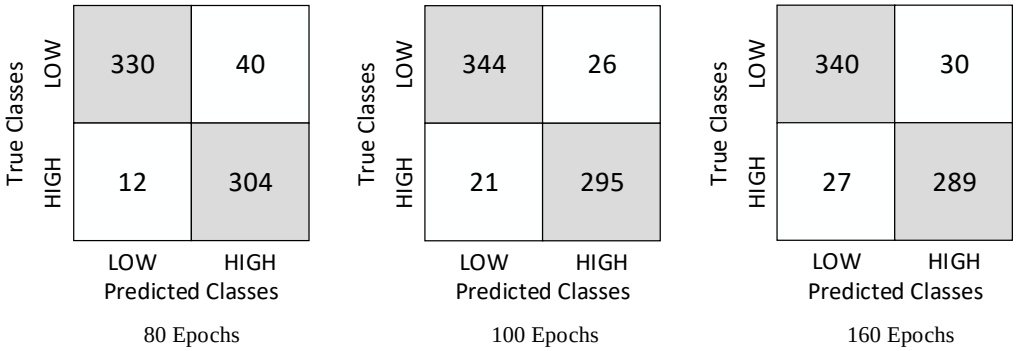

Figure 4. 11: AlexNet Model Confusion Matrices

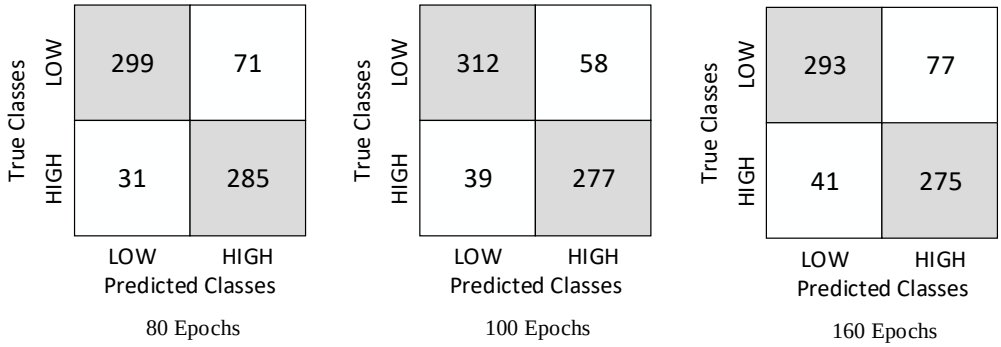

Figure 4.12: CNN Model Confusion Matrices

In conclusion, the results point to the efficacy of using AlexNet for the classification of sulfur-based concrete images based on porosity. While the simple CNN model employed also demonstrated a satisfactory performance, its lower accuracy suggests that there is room for improvement, perhaps by adjusting the architecture or fine-tuning the training parameters.

These findings pave the way for further research into the application of deep learning techniques in the field of civil engineering, particularly in the analysis and classification of construction materials. Future work could consider the incorporation of additional pre-trained models, larger datasets, or alternative deep learning techniques.

4.5 Comparative Analysis of Models on Independent Test Samples

To further assess the model's ability to generalize, an additional dataset including 100 sulfur-based concrete samples images was employed for the purpose of testing with saved models. The images indicated above, which differed from the initial set of 3,600 images used for training and testing, portrayed specimens of the same concrete samples.

The inclusion of these photographs was not incorporated at any point throughout the training or initial testing process. Among the entirety of the 100 photographs, 47 were identified as low porosity, while the remaining 53 were characterized as high porosity. The confusion matrix for independent test samples is shown in Figure 4.13 for CNN model, and the results of the performance metrics are shown in Figure 4.14.

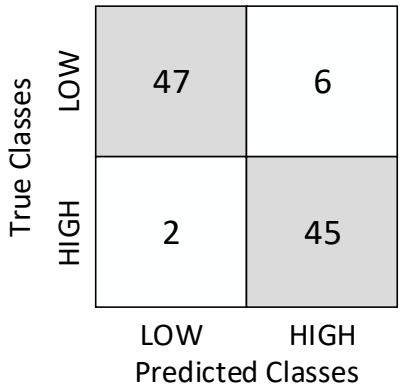

Figure 4.13: Confusion Matrix for Independent Test Samples of CNN model

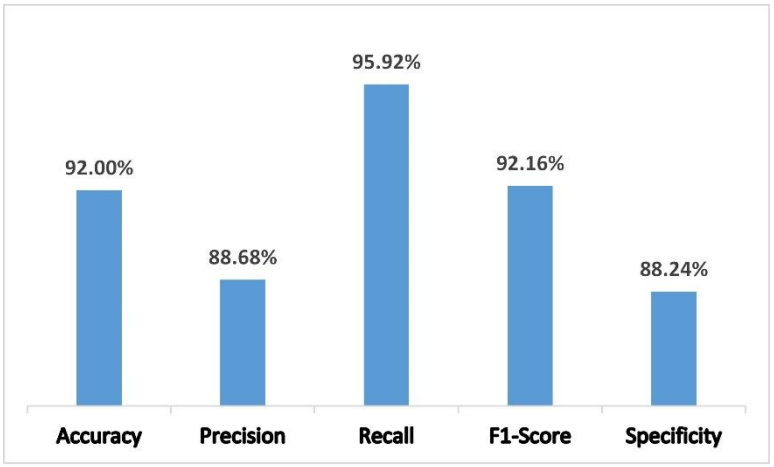

Figure 4.14: Metrics for CNN Model with Independent Test Samples

The second analysis was performed for AlexNet architecture. The confusion matrix for 100 test samples and performance results are shown in Figure 4.15 and 4.16 respectively.

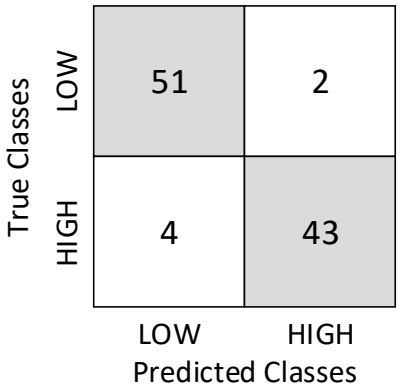

Figure 4.15: Confusion Matrix for Independent Test Samples of AlexNet model

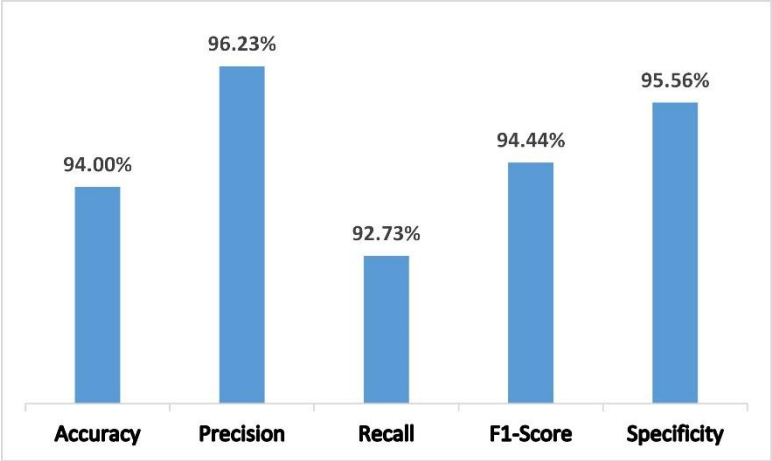

Figure 4.16: Metrics for AlexNet model with Independent Test Samples

4.6 Improving the Accuracy of Porosity Detection with Object Detection Algorithms

The existing methodology employed in the thesis has laid the groundwork for the examination of porosity through the utilization of image processing techniques. However, there is a wide range of possibilities for further improvement and expansion in this area. One area of inquiry that has promise for future investigation involves the incorporation of sophisticated object identification algorithms, such as YOLO (You Only Look Once), into our existing technique.

YOLO is a real-time object detection algorithm that is widely recognized for its proficiency in identifying objects in images with little latency. The utilization of YOLO can facilitate the expedited detection of voids inside concrete samples, thus augmenting the overall efficiency of the system. The YOLO architecture is specifically built to achieve a high level of precision. Using this technique has the potential to enhance the accuracy of void detection in images, thus facilitating more precise calculations of porosity.

Unlike conventional approaches that sequentially examine individual image regions, YOLO performs a holistic analysis of the entire image in a single iteration. This can be beneficial in situations where multiple voids are densely packed and can be identified simultaneously. Advanced object detectors such as YOLO reduce false positives by undergoing extensive training on large datasets. These detectors are highly proficient in distinguishing the object of interest from extraneous noise. Integrating the YOLO framework has the potential to mitigate false positives, thus enhancing the accuracy of porosity calculations by only considering genuine voids.

Future research should explore the possibility of enhancing the dataset by including a wider range of diverse photos depicting various concrete samples. This would result in an expanded dataset for YOLO's training, hence enhancing its robustness.

To cater to our specific application, it is necessary to perform fine-tuning on the YOLO model using a custom dataset consisting of concrete photographs. Although YOLO is initially pre-trained in a wide range of classes, this additional training is essential to optimize its performance for our specific task. The process entails fine-tuning the last layers of the YOLO network using our dataset, with the objective of enhancing its ability to reliably identify voids. Evaluation and validation are essential steps following the integration of YOLO, as they allow for a comprehensive assessment of the system's performance. An analysis of the outcomes obtained from the original system and the YOLO-integrated system would yield valuable insights regarding the enhancements accomplished.

In conclusion, the rapid evolution of the computer vision field necessitates the integration of cutting-edge algorithms such as YOLO, which can yield substantial

improvements in the performance of systems like ours. The incorporation of this technology has the potential to enhance accuracy and facilitate the development of real-time applications, such as on-site porosity detection in construction projects.

4.7 Limitations of The Study

This study offers significant contributions to the field of sulfur-based concrete sample categorization through the application of deep learning techniques. However, it is important to recognize the presence of certain inherent limitations in this research.

The investigation solely employed photos with a resolution of 1160 pixels by 2576 pixels. This particular resolution may fail to encompass intricate details that could prove essential in the determination of porosity. The potential for disparate outcomes exists when employing a higher resolution, while a lower resolution may fail to capture essential particulars. The potential for bias in the model may arise due to its dependence on a singular image size, particularly when considering practical scenarios that involve encountering images of diverse resolutions.

The process of image capturing through which photos of the concrete samples were obtained may potentially result in inconsistencies. When employing the Macro mode feature on a smartphone, it is possible to observe small discrepancies in lighting, angle, or focus among the captured photographs when maintaining a distance of 10 cm. The task of maintaining consistent lighting conditions for each sample can be difficult, even when using near-zero angle lighting to enhance the visibility of voids by making them appear darker. Even the slightest deviation has the potential to impact the lucidity and coherence of the visual representations, thus influencing the training process of the neural networks.

The method of slicing the samples at intervals of 5 cm may not fully encompass the entire range of porosity variations within an individual sample. Increasing the precision of the slicing intervals may result in obtaining more accurate and comprehensive images that effectively capture the variations in porosity within the sample. However, if the slicing process is not executed with a high level of precision, it may potentially result in the introduction of inconsistencies.

The investigation was carried out under a controlled setting, predominantly within the laboratory of the Civil department. This could potentially constrain the model's ability to make generalizations that are applicable to real-world situations that may exhibit varying conditions. For example, in the context of an outdoor construction environment, the samples under investigation could have been influenced by a range of environmental conditions that were not taken into account within the scope of this study.

Using smartphone cameras, as convenient as they may be, may not have the same precision as dedicated imaging devices due to hardware limitations. Variations arising from disparities in smartphone models, lenses, and software may produce discrepancies that are absent in the context of more standardized laboratory equipment.

Given the aforementioned constraints, it is important to exercise prudence when generalizing the outcomes of this investigation to wider scopes or alternative environments. Future research endeavors could potentially benefit from the inclusion of a broader spectrum of image resolutions, the implementation of more meticulously regulated imaging environments, and the potential integration of specialist imaging apparatus to ensure greater uniformity in outcomes.

Chapter 5

CONCLUSION

In this study, we investigated the performance of two deep learning models, a simple CNN and a pre-trained AlexNet model, for classifying sulfur-based concrete images based on porosity. We found that the AlexNet model achieved a higher accuracy rate (93.15%) than the simple CNN model (85.86%). This superior performance is likely due to the AlexNet model's pre-trained weights, which provide it with a better understanding of the underlying features of the images.

We also found that the optimal number of epochs for both models was 100. This suggests that training for longer periods of time does not necessarily lead to improved performance.

The results of this study suggest that deep learning has the potential to be a valuable tool for materials analysis in civil engineering. Future research should explore the use of additional pre-trained models and alternative deep learning approaches to further improve the performance of these models.

We believe that these findings are significant because they suggest that deep learning can be used to develop robust and accurate models for materials analysis. This has the potential to revolutionize the way that materials are characterized and evaluated and could lead to significant improvements in the design and construction of civil engineering structures.

Future research in this area should focus on developing deep learning models that are even more accurate and robust. This could be achieved by using larger and more diverse datasets, and by exploring alternative deep learning architectures.

We believe that the results of this study have the potential to make a significant contribution to the field of civil engineering. We hope that our findings will inspire other researchers to explore the use of deep learning for materials analysis.

REFERENCES

[1] N. J. Cassidy, R. Eddies, and S. Dods, “Void detection beneath reinforced concrete sections: The practical application of ground-penetrating radar and ultrasonic techniques,” *Journal of Applied Geophysics*, vol. 74, no. 4. Elsevier BV, pp. 263–276, Aug. 2011. doi: 10.1016/j.jappgeo.2011.06.003.

[2] Z. Chang, Z. Wan, Y. Xu, E. Schlangen, and B. Šavija, “Convolutional neural network for predicting crack pattern and stress-crack width curve of air-void structure in 3D printed concrete,” *Engineering Fracture Mechanics*, vol. 271. Elsevier BV, p. 108624, Aug. 2022. doi: 10.1016/j.engfracmech.2022.108624.

[3] Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6. Association for Computing Machinery (ACM), pp. 84–90, May 24, 2017. doi: 10.1145/3065386.

[4] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition.” arXiv, 2015. doi: 10.48550/ARXIV.1512.03385.

[5] Y. Fujita, Y. Mitani, and Y. Hamamoto, “A Method for Crack Detection on a Concrete Structure,” *18th International Conference on Pattern Recognition (ICPR’06)*. IEEE, 2006. doi: 10.1109/icpr.2006.98.

[6] G. Salamunićcar, S. Lončarić, P. Pina, L. Bandeira, and J. Saraiva, “Integrated method for crater detection from topography and optical images and the new

PH9224GT catalogue of Phobos impact craters," *Advances in Space Research*, vol. 53, no. 12. Elsevier BV, pp. 1798–1809, Jun. 2014. doi: 10.1016/j.asr.2013.11.006.

[7] Barati Farimani, M. Heiranian, and N. R. Aluru, "Identification of amino acids with sensitive nanoporous MoS2: towards machine learning-based prediction," *npj 2D Materials and Applications*, vol. 2, no. 1. Springer Science and Business Media LLC, May 24, 2018. doi: 10.1038/s41699-018-0060-8.

[8] M. Flah, I. Nunez, W. Ben Chaabene, and M. L. Nehdi, "Machine Learning Algorithms in Civil Structural Health Monitoring: A Systematic Review," *Archives of Computational Methods in Engineering*, vol. 28, no. 4. Springer Science and Business Media LLC, pp. 2621–2643, Jul. 29, 2020. doi: 10.1007/s11831-020-09471-9.

[9] J. Helal, M. Sofi, and P. Mendis, "Non-Destructive Testing of Concrete: A Review of Methods," *Electronic Journal of Structural Engineering*, vol. 14, no. 1. EJSE International, pp. 97–105, Jan. 01, 2015. doi: 10.56748/ejse.141931.

[10] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep Hypersphere Embedding for Face Recognition." arXiv, 2017. doi: 10.48550/ARXIV.1704.08063.

[11] Y. Liu and J. K. W. Yeoh, "Automated crack pattern recognition from images for condition assessment of concrete structures," *Automation in Construction*, vol. 128. Elsevier BV, p. 103765, Aug. 2021. doi: 10.1016/j.autcon.2021.103765.

[12] Arbaoui, C. Aribi, M. Boumaiza, S. Mohamadi, and F. A. Ahmed, “CNN-based Concrete Cracks Detection Using Multiresolution Analysis,” *2022 7th International Conference on Image and Signal Processing and their Applications (ISPA)*. IEEE, May 08, 2022. doi: 10.1109/ispa54004.2022.9786328.

[13] S. Dorafshan, R. J. Thomas, and M. Maguire, “Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete,” *Construction and Building Materials*, vol. 186. Elsevier BV, pp. 1031–1045, Oct. 2018. doi: 10.1016/j.conbuildmat.2018.08.011.

[14] H. Kim, E. Ahn, M. Shin, and S.-H. Sim, “Crack and Noncrack Classification from Concrete Surface Images Using Machine Learning,” *Structural Health Monitoring*, vol. 18, no. 3. SAGE Publications, pp. 725–738, Apr. 23, 2018. doi: 10.1177/1475921718768747.

[15] P. Chopra, R. K. Sharma, M. Kumar, and T. Chopra, “Comparison of Machine Learning Techniques for the Prediction of Compressive Strength of Concrete,” *Advances in Civil Engineering*, vol. 2018. Hindawi Limited, pp. 1–9, 2018. doi: 10.1155/2018/5481705.

[16] G. Bianco, A. Gallo, F. Bruno, and M. Muzzupappa, “A Comparative Analysis between Active and Passive Techniques for Underwater 3D Reconstruction of Close-Range Objects,” *Sensors*, vol. 13, no. 8. MDPI AG, pp. 11007–11031, Aug. 20, 2013. doi: 10.3390/s130811007.

[17] S. Wolter, F. A. H. Uhre, M. T. Hasholt, V. A. Dahl, and F. Anton, “Air void analysis of hardened concrete by means of photogrammetry,” *Construction and Building Materials*, vol. 226. Elsevier BV, pp. 953–964, Nov. 2019. doi: 10.1016/j.conbuildmat.2019.07.203.

[18] B. Biskup, H. Scharr, U. Schurr, and U. Rascher, “A stereo imaging system for measuring structural parameters of plant canopies,” *Plant, Cell & Environment*, vol. 30, no. 10. Wiley, pp. 1299–1308, Oct. 2007. doi: 10.1111/j.1365-3040.2007.01702.x.

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation.” arXiv, 2013. doi: 10.48550/ARXIV.1311.2524.

[20] M., Cordts, M., Omran, S., Ramos, T., Rehfeld, M., Enzweiler, R., Benenson , U., Franke , S., Roth, and B., Schiele, “The Cityscapes Dataset for Semantic Urban Scene Understanding.” arXiv, 2016. doi: 10.48550/ARXIV.1604.01685.

[21] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, “Deep Learning for Content-Based Image Retrieval,” *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, Nov. 03, 2014. doi: 10.1145/2647868.2654948.

[22] C. Lee and J. Hogan, “Automated crater detection with human level performance,” *Computers & Geosciences*, vol. 147. Elsevier BV, p. 104645, Feb. 2021. doi: 10.1016/j.cageo.2020.104645.

[23] O. Yaman, N. Hearn, and H. M. Aktan, “Active and non-active porosity in concrete Part I: Experimental evidence,” *Materials and Structures*, vol. 35, no. 2. Springer Science and Business Media LLC, pp. 102–109, Mar. 2002. doi: 10.1007/bf02482109.

[24] B. Chen, X. Zhang, R. Wang, Z. Li, and W. Deng, “Detect concrete cracks based on OTSU algorithm with differential image,” *The Journal of Engineering*, vol. 2019, no. 23. Institution of Engineering and Technology (IET), pp. 9088–9091, Oct. 31, 2019. doi: 10.1049/joe.2018.9191.

[25] Y. Wei, Z. Wei, K. Xue, W. Yao, C. Wang, and Y. Hong, “Automated detection and segmentation of concrete air voids using zero-angle light source and deep learning,” *Automation in Construction*, vol. 130. Elsevier BV, p. 103877, Oct. 2021. doi: 10.1016/j.autcon.2021.103877.

[26] N. Zhang, R. Farrell, F. Iandola, and T. Darrell, “Deformable Part Descriptors for Fine-Grained Recognition and Attribute Prediction,” *2013 IEEE International Conference on Computer Vision*. IEEE, Dec. 2013. doi: 10.1109/iccv.2013.96.

[27] F. N. Iandola, A. Shen, P. Gao, and K. Keutzer, “DeepLogo: Hitting Logo Recognition with the Deep Neural Network Hammer.” arXiv, 2015. doi: 10.48550/ARXIV.1510.02131.

[28] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig, “From Captions to Visual Concepts and Back.” arXiv, 2014. doi: 10.48550/ARXIV.1411.4952.

[29] S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN Features off-the-shelf: an Astounding Baseline for Recognition.” arXiv, 2014. doi: 10.48550/ARXIV.1403.6382.

[30] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition.” arXiv, 2014. doi: 10.48550/ARXIV.1409.1556.

[31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision.” arXiv, 2015. doi: 10.48550/ARXIV.1512.00567.

[32] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition.” arXiv, 2015. doi: 10.48550/ARXIV.1512.03385.

[33] H. Zhang, Y. N. Dauphin, and T. Ma, “Fixup Initialization: Residual Learning Without Normalization.” arXiv, 2019. doi: 10.48550/ARXIV.1901.09321.

[34] Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I, Polosukhin, “Attention Is All You Need.” arXiv, 2017. doi: 10.48550/ARXIV.1706.03762.

[35] K. Shukla and A. Gupta, “Mix Design and Factors Affecting Strength of Pervious Concrete,” *Lecture Notes in Civil Engineering. Springer Singapore*, pp. 125–139, Jun. 13, 2019. doi: 10.1007/978-981-13-7615-3_11.