High Voltage Insulator Real-time Condition Classification

Capstone Report

Askar Gylymbekov

Nazarbayev University
Department of Electrical and Computer Engineering
School of Engineering and Digital Sciences

This project report was created on TexStudio editing platform using LaTeX. All the figures were drawn using draw.io online software tool.

Electrical and Computer Engineering
Nazarbayev University
http://www.nu.edu.kz

Title:
High Voltage Insulator Real-time Condition Classification

Theme:
High Voltage Insulation

Project Period:
Fall 2023

Project Group:

Participant(s):
Askar Gylymbekov

Supervisor(s):
Mehdi Bagheri

Copies: 1

Page Numbers: 35

Date of Completion:
April 26, 2024

Abstract:

Insulators have a dual purpose of mechanically supporting and electrically isolating live phase conductors from the support tower in power systems. Due to experiencing harsh weather conditions, insulators may become contaminated or damaged. As a result, electrical and mechanical properties of insulator may deteriorate. Thus, it is significant to automize the process of condition classification of High Voltage insulator to prevent accidents and therefore ensuring secure service of transmission lines. This paper examines the effect of pollution at insulator's surfaces to the Dielectric Dissipation Factor and therefore research was conducted in 2 different disciplines: 1) Calculation of dielectric dissipation factor using Ansys Maxwell software; 2) High voltage insulators contamination type classification using Convolutional Neural Networks. Ansys Maxwell software is used to simulate the Dielectric Dissipation Factor of insulator under different types of contamination. In this part, three contamination levels (light, medium and heavy) will be considered which include soil, cement, iron, calcium and aluminum as pollutants. The results show trend in which DDF increases with an increase of contamination level. In image classification part, Convolutional Neural Networks will be used to create a classifier. Insulators will be classified between 4 types and these are: clean, contaminated with soil, contaminated with water and contaminated with snow. Validation set assessment results in 91% of accuracy.

Contents

Preface

Electricity, nowadays, for human being is one of the basic needs along with food, water, air and shelter. Power network is complex system which includes generation, transmission and distribution of electricity to people. Thus, it is vital part of infrastructure which requires monitoring and maintenance. Insulators are used in all the branches of power system which objective is to insulate the electricity in such a way to prevent the unwanted flow leading to the power losses and ultimate economical issues. Nowadays, insulators are exposed to harsh environment which often leads to contamination of insulators which in its turn may lead to deterioration of dielectric properties of insulators which should be prevented at all costs. This research was conducted in two different fields namely dielectric dissipation factor simulation using Ansys Maxwell and Convolutional Neural Networks to obtain methods which could be used for further research and to become one of the central points in security of power systems.

I would like to express my sincere gratitude to my supervisor, professor Mehdi Bagheri who was patient with me all the time. I sincerely appreciate all the work he had done for this project, giving me inspiration and discipline during the research process. Many thanks to Arailym Serikbay, PhD candidate, who helped me with concepts and mechanisms behind the operation principles of insulators. Special thanks to Sadjad Shafiei, research assistant at Nazarbayev University, who gave me valuable guidance in Ansys Maxwell. Without their help, this project would not have been of good quality nor completed on time.

Nazarbayev University, April 26, 2024

Askar Gylymbekov
<askar.gylymbekov@nu.edu.kz>

Chapter 1

Introduction

Electrical power distribution networks is one of the most important inftastructure objects nowadays[1]. Maintaining the integrity of system is essential part of ensuring continuous power supply to consumers. As was mentioned above insulators have dual purpose in transmission lines that is mechanically supporting and electrically isolating. Thus the integrity of transmission lines and therefore continuity of power supply massively depends on conditions of insulators. However as they are exposed to the harsh nature, their electrical and mechanical properties worsen. According to [2], composite insulators are inclined to accelerated aging in polluted environments. Furthermore, environmental conditions like moistures and temperature changes may lead to arcs and sparks in insulating material[3].

Assessment of insulator's dielectiric property could be done via studying its dielectric dissipation factor. Measurement of the DF, also known as tan-delta, was created in the last 10 years and considered as significantly effective method for insulation assessment of capacitive equipment[4].The reciprocal of the ratio of the capacitive reactance to the resistance of the insulating material at a certain frequency is known as the dissipation factor. It gauges an insulating material's inefficiency. Basically increase of dissipation factor means deterioration of insulation leading to potential flashovers which is one of the most important faults in high voltage line insulators[5].

Pollution layers are found on the surface of insulators. Pollutants are generally salt, dust, water, and industrial wastes. These layers often lead to the alterations in electrical properties of insulators as they are in contact with contamination leading to the development of leakage current[6], and as a result increase of dielectric dissipation factor happens.

As it was written electrical power systems' dependability and safety are seriously threatened by insulator contamination. For these systems to be maintained and operated effectively, it is essential to automize the process of insulators' contamination type identification which is one of two aims of this research. In this study, we also aim to create a unique method for automatical classification of contamination types based on picture using Convolutional Neural Networks (CNN).

Many environmental elements, including dust, salt, pollution, and other particulate matter, can contaminate an insulator. High-resolution photos may be used to capture the distinctive visual features of each type of contamination, which appear as textures or patterns on the insulator surface. Our goal is to use CNNs' capacity to use these visual features to reliably and precisely categorize various forms of contamination.

Nowadays, assessment of outdoor high voltage insulators is performed using special equipment which requires insulators to be disconnected from the transmission line which is undoubtedly impractical and inefficient way. Thus, in this research, emphasis was made on using software to model an insulator and simulate the effect of pollution on dielectric dissipation factor. Therefore, completely similar to the real, a high voltage insulator was modelled in Ansys Maxwell software. Modelled insulator was artificially contaminated with different levels of pollution and simulations were carried to determine the values of dielectric dissipation factor caused by presence of contaminations. These results were then used to relate them with contamination level. To accelerate the process of condition classification of insulator, it is crucial to be able to automize the process of contamination determination of insulators, thus, in this research, insulators' contamination type classifier was built using CNN. Images of insulators were captured in the laboraty of Nazarbayev University and classifier was trained on these images of insulators and was able to predict between for types of insulators: clean, contaminated with soil, contaminated with water and contaminated with snow.

Chapter 2

Background

2.1 Dielectric Dissipation Factor

To work with Ansys Maxwell, the deep knowledge of the software, as well as a study of other researches carried in this theme is required. The knowledge of methods which are used in software is an essential part.

To simulate the electric field around insulators, researchers use numerical methods such as the boundary element method (BEM) [6] or finite element method (FEM) [7] with QuickField [8], COMSOL Multiphysics [9], Maxwell [10] and ANSYS software [11]. These methods involve dividing the insulator surface into small elements or boundary segments and solving the equations that describe the behavior of the electric field in each element or segment.

In [8], the authors discuss the problems associated with contamination on overhead transmission line insulators and the uneven distribution of voltage and electric field, which can lead to corona, partial discharge, premature ageing, and flashover. The FEM software (QuickField) was used to calculate the distribution of voltage and electric field on the surface of glass insulators, specifically cap and pin insulators, with and without contaminants. This study analyzes a model of a U40B high-voltage (HV) glass suspension insulator. The model also includes a thin film covering the insulator's entire surface and is modeled as a uniform conductive layer with a 2mm thickness, representing contamination from seawater. The simulation is performed in an axisymmetric 2D problem with AC conduction analysis and a 50Hz frequency. Gauss's Law, Current Continuity Equation, and Ohm's Law are used in the analysis. The simulation is modeled in free space with a potential of 11kV applied to the insulator pin, and the insulator cap is grounded. The simulation results for clean insulators show that the voltage gradually decreases from pin to cap, while the e-field is highest inside the glass material, near the pin, and around the cap. For contaminated insulators, the voltage is evenly distributed from pin to cap along the shed, while the e-field is concentrated within the glass between the cap and pin, followed by the shed. The density of field lines indicates the strength of the e-field inside and around the insulator. This study found that the voltage distribution is unstable and decreases significantly at the insulator ribs for both clean and contaminated insulators, emphasizing the need for careful attention in designing the insulator, especially at the ribs.

The relationship between surface conductance and the electric field was investigated in [9]. A new method for calculating electric field distribution over contaminated polymeric insulators based on electrical characteristics was introduced here. The method was tested on two different 33 kV polymeric insulator profiles using conductance testing and leakage current measurements under different conditions. The FEM was used to compute the electric field and current density distributions along the insulator. The COMSOL Multiphysics software and finite element modeling were used to analyze the effect of conductivity on the electric field and current density distribution of 33-kV polymeric insulators with and without contamination. The insulator's geometry is designed, and the quasi-static electric field is employed for numerical analysis. The text describes an experiment and simulation conducted to study the electrical properties of two types of insulators under wetting and drying conditions. The breakdown voltage threshold was found to be less than 10 kV/cm under uniform wetting. A proposed model showed a distribution of electric field and current density with peaks on the surface of the insulators, and local stress increases were observed due to applied rain wetting. The insulator profile was found to have an influence on the electrical properties.

In [12], a research study utilized COMSOL Multiphysics to examine how various degrees of pollution affect the distribution of electric fields in polymeric insulators. The study considered four pollution levels (light, medium, heavy, and very heavy) and analyzed their impact on the electric field and potential distribution. The simulation modelled the pollution as a uniform conductive layer of water on the surface of the insulator, and different conductivity values were used to analyze the electric field. The results revealed that the electric field strength is highest at the junction of the sheath and shed and lowest at the top of the shed. Also found that electric field stress increases with higher pollution levels. Another research paper investigates the impact of different types of coating damage on the performance of 33 kV porcelain insulator strings in both polluted and clean conditions. The insulators were coated with room-temperature vulcanizing (RTV) and tested with both partially damaged and undamaged coatings [13]. The findings indicate that the electrical characteristics of the RTV-coated insulators are significantly affected by the distribution of coating damage and the level of humidity. The electric field and potential difference were analyzed through the use of the FEM, and it was observed that various combinations of coating damage, pollution, and humidity resulted in changes to the resistance of the surface pollution layer and uneven electric field distribution. The study concludes that the parameters of coated insulators are closely linked to the distribution of coating damage on the surface of the insulators, particularly when humidity is present.

The electric field distribution along a polymeric insulator affected by wet and contaminated conditions is investigated in [12]. The study uses simulations with COMSOL Multiphysics to examine how the electric field is impacted by the conductivity and thickness of the pollution layer. The pollution layer is represented as a conductive water layer with conductivity and thickness measures. The results of the study show that as the conductivity and thickness of the pollution layer increase, the electric field intensity also increases. In [14], researchers used FEM analysis to examine the effects of acid precipitation on an outdoor HV glass insulator string designed using AutoCAD. The study analyzed the insulator using COMSOL Multiphysics at a constant voltage of 110kV and evaluated the impact of pH levels ranging from 2.5 to 5.5 on electric field distribution and heat generation. The results showed that high electric fields and increased heat generation occurred at lower pH levels, and these findings can be useful in laboratory experiments and field tests for exploring the performance of HV insulators in areas with heavy industry and exposed to acid rain pollution. The electric field properties of energized insulators can significantly impact surface contamination, but there is little information on how electric fields influence pollution deposition. Dongdong et al conducted a study that establishes a model of a three-unit XP-160 insulator string, simulating the deposition process of pollution particles using the finite element method [15]. The study found that the electric field changes the vertical speed of pollution particles, accelerating their deposition process. The ratio of the particle capture coefficient can be used to reflect the degree of pollution on DC-energized and non-energized insulators, with the study finding a ratio like a field measurement result.

The method of finite elements entails partitioning a structure into smaller components for evaluation and subsequently merging them to analyze the structure in its entirety. To simulate and compute the distribution of the electric field, the process involves discretization, specifying electrical properties for each component, applying Maxwell's equations and boundary conditions to interconnect components, and then selecting an appropriate computation technique to solve for unknown points [10]. This study found that the electric field distribution on insulator strings without a corona ring relies on the insulator material, profile, and configuration of the corona ring. Accordingly, to achieve the most significant decrease in the electric field for a specific type of insulator, the parameters of the corona ring should be optimized. Additionally, it was determined that the relationship between the electric field and the insulator material and profile becomes less significant as the distance from the insulator string increases.

In [11], the researchers focus on the impact of lightning on power line failures in Malaysia, which are often caused by pollutants on the surface of polymer insulators. The research investigates the potential benefits of using RTV coating material to enhance the electrical performance, electric field distribution, and voltage profile of polymer insulators. The study employs ANSYS software based on the U50 value and demonstrates that the application of RTV coating material can increase the voltage breakdown level of insulators and decrease the electric field concentration at the triple region. The study found that applying an RTV coating can improve the voltage withstand capabilities of a polymer insulator under a lightning impulse.

2.2 Convolutional Neural Networks

Convolutional neural networks, having a 45-year history of application for visual tasks[16], is an exceptional type of deep learning models that have revolutionized a number of fields, including computer vision and medical imaging via processing visual input [17]. CNNs is characterized by its capability to automatically extract hierarchical feature representations from unprocessed pixel data, which allows them to recognize intricate patterns and structures in pictures[18].

In [17] offers a thorough analysis of deep learning, including its theoretical basis, historical evolution, and real-world applications. The study shows the development path of deep neural network, emphasizing significant developments like recurrent neural networks (RNNs) and convolutional neural networks (CNNs). It explores the fundamentals of deep learning, including regularization strategies, optimization methods, and network architectural design. Additionally, the study looks at deep learning's uses in a number of fields, such as speech recognition, computer vision, and speech processing. It highlights importance of large-scale datasets and computer power in efficient training of deep neural networks.

In paper [19], AlexNet architecture described, which is a Convolutional Neural Network model that showed significant performance in the ImageNet Large Scale Visual Recognition Challenge(ILSVRC) in 2012. Paper is devoted to addressing image classification challenge via using CNN (part of Deep learning). Architecture of the model contained multiple convolutional and pooling layers which are followed by fully connected layers. The model in comparison with other learning methods surpassed others via using large datasets, showing advanced regularization. Overall, this paper shows the efficiency of CNN in comparison with other learning techniques.

GoogleNet - a CNN model. In this paper, an novel approach in network design is proposed which is achieved by inventing so called "inception model" which operate based on parallel convolutional operations with different sizes of kernel. Using this architecture resulted in breakthrough performance in image classification, maintaining computational efficiency and decreasing the risk of overfitting. In addition, paper also includes researches in auxiliary classifiers and global average pooling to enhance gradient flow during training process[20].

In [21],the use of VGG16 Convolutional Neural Network model improved with transfer learning is explored. The model categorizes magnetic resonance images of the brain into healthy and unhealthy types. The study shows the advantages of transfer learning which enables adapt the pre trained model to a new small dataset avoiding the problem of overfitting

Chapter 3

Methodology

3.1 High-voltage insulator modelling

The glass insulator shown in Fig. 3.1 was modelled in ANSYS Maxwell software. The components this insulator has are shown in Fig. 3.2, its the names and the materials it is made of and their relative permittivities are written in Table 3.1.

Figure 3.1

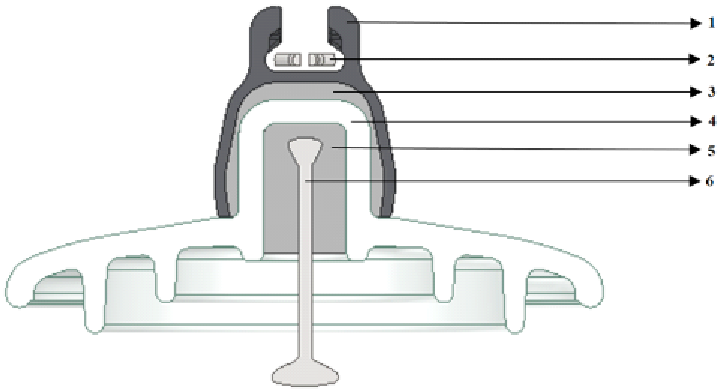


Figure 3.2: Components of insulator

Table 3.1: The material type and relative permittivity of the glass insulator's components

	Name	Material	Relative permittivity
1	Iron cap	Cast iron	1
2	Locking key	Steel stainless	1
3	Cement	Cement	9.798
4	Insulating glass	Glass	5.5
5	Cement	Cement	9.798
6	Steel pin	Cast iron	1

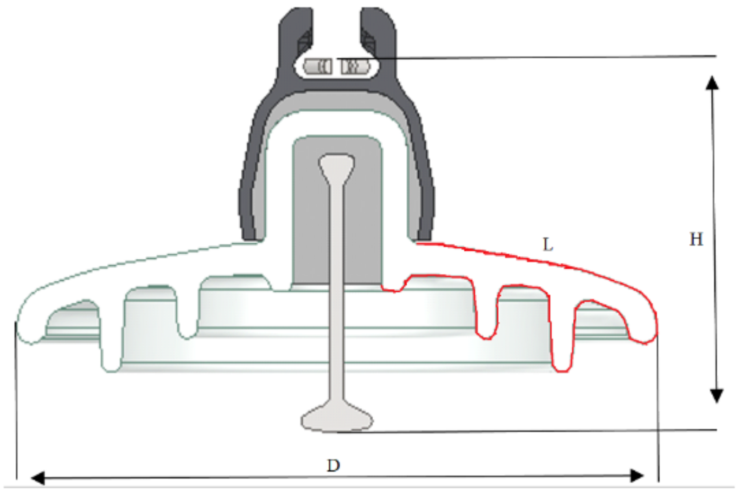


Figure 3.3: The shape and the structure of the glass insulator

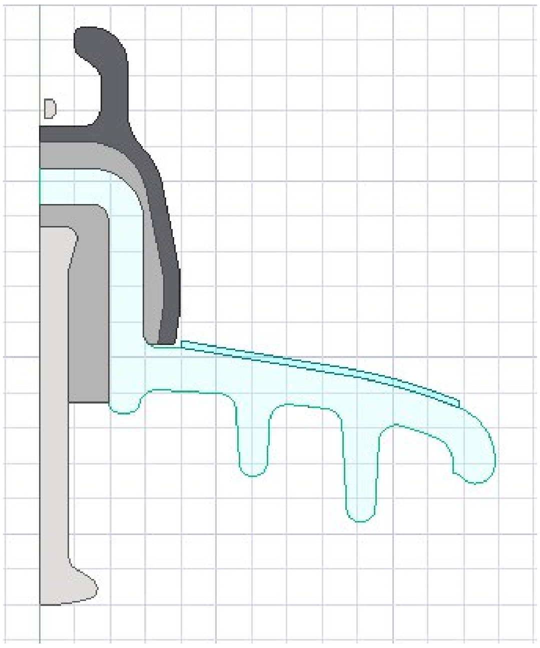


Figure 3.4: 2D model of insulator

Table 3.2: Values of specific parameters of insulator

Parameters	D	L	H
Values	258 mm	390 mm	335 mm

In Fig. 3.3, the specific parameters of the glass insulators are shown, and their values are recorded in Table 3.2.

The pollution was layered on the surface of the insulator covering insulating glass. Pollutants as mentioned above include salt, dust, water, and industrial wastes. Basically they were used to cover the surface of insulating glass separately (not mixed).

3.2 Dielectric Dissipation Factor

To conduct simulations Ansys software was used which is a software program designed to perform finite element modeling, and is capable of numerically solving a diverse range of problems. These problems cover a wide range, including heat transport, fluid mechanics, static and dynamic analysis, structural analysis as well as acoustic and electromagnetic phenomena [22]. In this research, Ansys Maxwell was used which allows the analysis of different electromechanical systems. It is

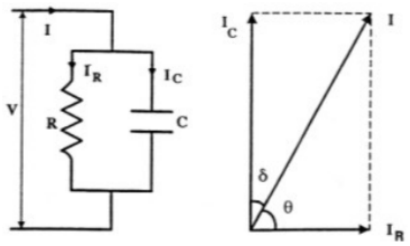


Figure 3.5: Equivalent circuit of insulator and phasor diagram [24]

capable of finding electromagnetic fields via the usage of finite element analysis [23]. In this research, we will try to apply the DDF test also known as the loss angle test. It is a non-destructive test which is used to determine the insulation quality. Commonly, in AC systems insulators are replaced with an equivalent circuit which consists of a resistor and capacitor connected in parallel as shown in Figure 3[24].

In this scenario, DF is calculated as a relationship between resistive current and capacitive current. However, expressing currents via Ohm's will give us an expression according to which the DF will be related to the angular frequency, resistance and capacitance of the system. Thus, we are required to obtain the capacitance and resistance of the insulator. Aging or contamination of insulator results in decrease of resistance thus increasing the real current which basically means increase of DF [25].

3.3 Simulation of DDF

3.3.1 Finding capacitance

Capacitance is the ratio of electric charge to voltage. This formula will be the basis for the following calculations. Moreover, the cap and pin are considered as plates of a capacitor, while glass is a dielectric between them and the distance between plates is to be taken as L or creepage distance. It is important to note that cement was not taken into account in these calculations, thus only glass as a dielectric is considered. The steps to obtain capacitance are as follows: 1) to assign floating charges to the cap and pin 2) to create a polyline shown in Fig. 3.3 as L 3) utilization of the calculator in Ansys, the potential difference over the creepage distance will be calculated. Thus obtaining an electric field plays an important role. Capacitance will be the relationship between applied charges at the pin and cap to the potential difference between those points

3.3.2 Finding DDF for different pollution degrees

The DDF determination process is performed in the following way: 1) Creation of 2D model of insulator shown in Fig. 3.4. The layer of contamination could be seen on Fig. 3.4. 2) Materials for components of insulator are assigned according to the Table 3.1. 3) To create region outside the insulator and assign boundary for solution region (region-select edges-assign boundary-baloon). 4) Assign solution type as AC conduction and choose geometry mode as cylindrical about Z. 5) To assign excitations: pin is at 11 KV and cap is grounded. 6) To assign matrix for simulation(parameters-assign-matrix). 7) In analysis, to add the solution setup(maximum number of passes: 10 and percent error: 8) To analyze the setup. 9) As simulation is done, to obtain the conductance and capacitance values in Results by clicking on solution data. This process was performed 19 times for different types of pollution profile and different types of contaminants. The contamination levels were taken as 2 mm, 4 mm, and 6 mm. The main parameters used to calculate DDF were conductance (G) and capacitance (C) of insulator. Moreover, pollution mass as well as pollution surface density were calculated based on the following formula respectively.

$$m = \rho * V \tag{3.1}$$

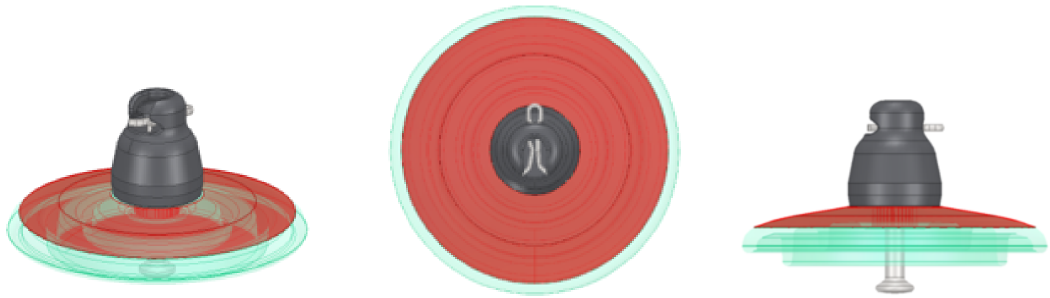
$$\sigma = m/S \tag{3.2}$$

In Figure 3.6 the pollution regions of insulator are shown in different perpectives.

Table 3.3: Pollution profiles

Pollution level	Thickness (mm)	Volume (mm$\hat{3}$)	Surface area (cm$\hat{2}$)
Light	2	78836.38	40.606
Medium	4	157672.76	40.606
Heavy	6	236509.13	40.606

To have a visual understanding of relationship between contamination profile and DDF value, it was decided to determine the pollution mass and surface density of pollution. In order to find the mass of pollution Eq. 3.1 the following steps were accomplished: 1) To create the 3D model from 2D via rotation it around Z axis for 360 degrees. 2) To select the volume contamination as an object and via measure tool find the volume of contamination 3) Find the mass density of contaminant material properties (solution type needs to be changed to transient). 4) To use Eq. 3.1 to find mass In order to find surface density of pollution, the following steps are done: 1) In 3D model change the select mode to Face 2) To select required surface and measure it via tool 3) To use Eq. 3.2 to find the surface density of pollution. These steps were done for all types of contamination and pollution profiles The contamination profiles are shown in Table 3.3. Obtained results are shown in Tables 4.1, 4.2, 4.3 in the Chapter 4



(a) Trimetric view **(b)** Top view **(c)** Front view

Figure 3.6: Contamination areas of insulator

3.4 Insulator contamination type classifier

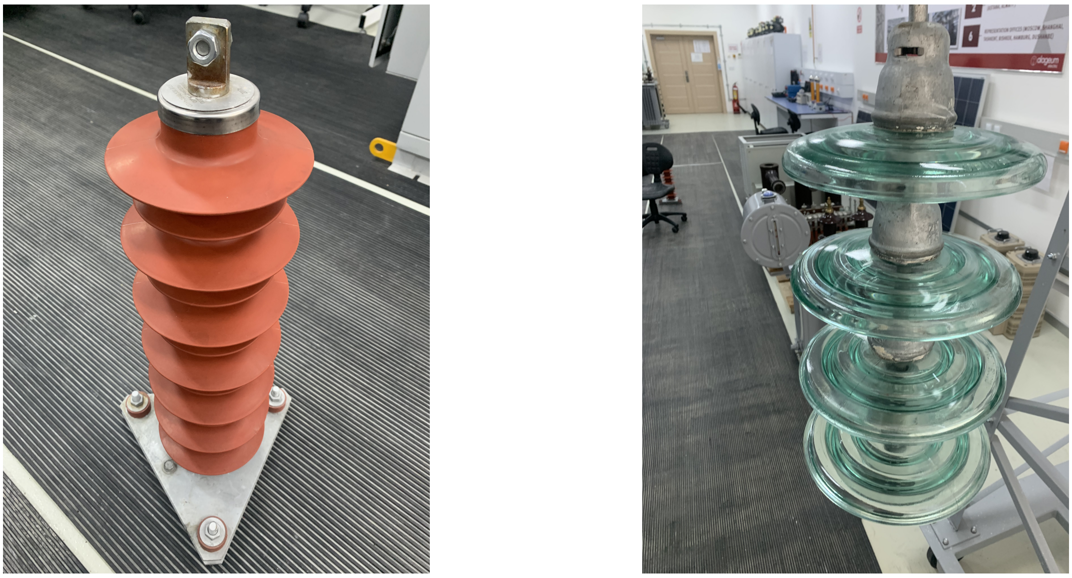
Creation of machine learning model is complex process which requires systematic approach. In this section I provide readers with detailed overview of steps which were taken in order to create model.

3.4.1 Data Collection

The process of building an effective image classification model starts with an acquisition of high-quality and diverse dataset. The labeled dataset is found the be critical in image classification For this research, high-quality pictures, clearly showing different types of contamination were required. Contaminations are usually located at the surfaces of insulators, thus in laboratory I tried to replicate situation of contamination with different contaminants namely: soil, water and snow.

Dataset contained 2315 pictures which were divided into 3: 1820, 455, 40 as training set, validation set and for prediction respectively. Insulators used for image acquisition were of 2 different types: polymer made and glass made. Pictures were captured from different angles to ensure the diversity of dataset and more efficient way prediction.

Image acquisition was conducted using smarthone "Iphone XR" with 12 megapixel camera which basically means that each picture had 4000 by 3000 pixels resolution. Examples of images captured are shown in Figures 3.7, 3.8, 3.9 and 3.10

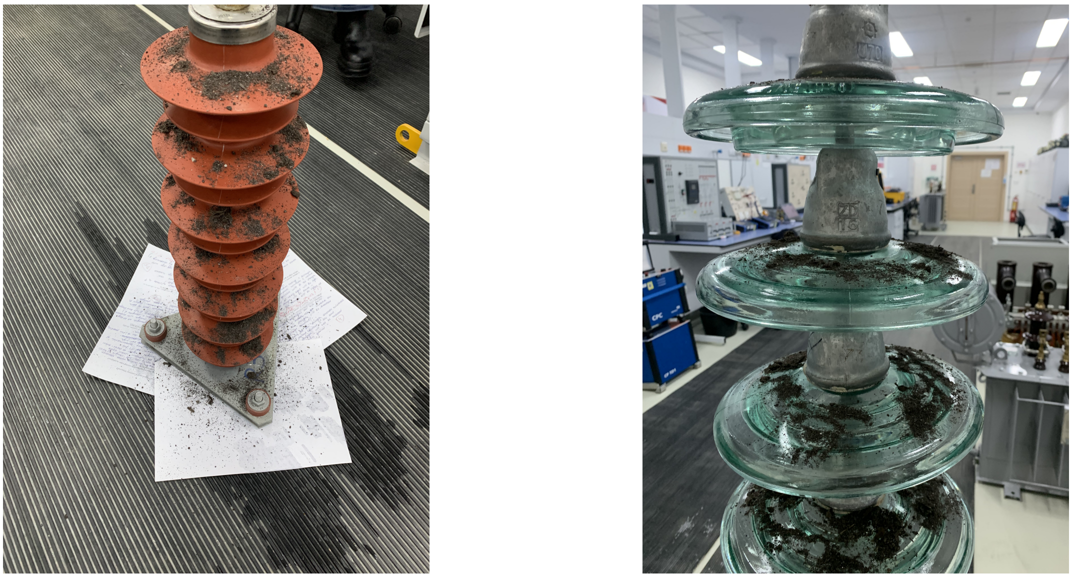


(a) Polymer insulator **(b)** Glass insulator

Figure 3.7: Clean insulators

(a) Polymer insulator **(b)** Glass insulator

Figure 3.8: Insulators contaminated with snow



(a) Polymer insulator **(b)** Glass insulator

Figure 3.9: Insulators contaminated with soil

(a) Polymer insulator **(b)** Glass insulator

Figure 3.10: Insualors contaminated with water

3.4.2 Convolutional Neural Network

The selection of appropriate tool to build the image classifier is probably the most important part of whole research. In this regard, it was undetstood that Convolutional Neural Network has the most promising and probably efficient way of image feature extraction compared to others. Thus, the CNN was chosen to be apparatus for following deep-learning model.

3.4.3 Jupyter Notebook, Tensorflow and Keras

Environment where the model will be implemented is Jupyter Notebook which is open-source web application which could be reached with any browser thus making it easy to access. Moreover, it gives an opportunitiy to divide the whole code in the different parts, with multitask cell that allow you to write not only code but also notes and others texts. For coding, Python was chosen as a language.

In order to create the model, various libraries were installed and used: 1) Tensorflow 2) Keras. Tensorflow. Both of them are modules designed to build machine learning models, however in terms of functionalities Tensorflow is more diverse as it provides both low level and high level APIs. In fact nowadays, Keras is integrated into Tensorflow. Keras on the other hand is useful in terms of convenience, as its functionality is user-friendly.

3.4.4 Steps

Firstly the images were divided according to its contamination type into 4 different folders: 1) clean 2) contaminated with snow 3) contaminated with soil 4) contaminated with water. Dataset was then uploaded to Jupyter notebook for further work with it.

Secondly, new Jupyter notebook file is created, and all the necessary libraries are firstly "pip-intalled" and imported to the notebook. Firstly importing "Tensorflow" as "tf" and then from "Tensorflow" Keras is imported. From Keras, "Sequential" and "layers" classes (building blocks) are imported.

Thirdly, I used image_dataset_from_directory function to create a dataset from a directory which was previously uploaded to Jupyter Notebook. Hyperparameters of function were set in such a way that 20% of dataset will be used as validation set while other 80% for training. I also assigned "seed" parameter to avoid random variations in dataset. The function was used two times with "training" subset and "validation" subset. It is important to note that image size was assigned to 180x180 pixels meaning that images were resized to this scale. Moreover, batch size was assigned to 32 meaning that in one iteration, model will be trained with 32 images, leading to the fact that overall in each training we had 57 iterations.

As it could be seen from Figure A.6, the nested loop was created in order to conduct model selection. Because scikit learn do not support tensorflow operations, it was decided that model selection will be conducted based on variations of hyperparameters of convolutional layers such as: number of filters and kernel size. In our architecture, there are 3 convolutional layers, the number of filters in each layer is varied between 16, 32 and 64, moreover, the kernel size is also varied betwenn 3x3, 5x5 and 7x7. Overall, this nested loop results in training of 81 models. As a result of which the model with following hyperparameters was found as most accurate: 32 filters in first convolutional layer, and 16 filters in both second and third convolutional layers and with the kernel size equal to 3 by 3 with a validation accuracy near 94%.

Fourthly, we create model architecture, with classes "Sequential" and layers. In architecture, as seen in Fig. A.4, 1 input layer, 3 hidden layers and 1 output layer. Input layer is preprocessing layer where picture is rescaled or normalized in order to make the process of training easier, moreover input shape of images and depth are defined in the hyperparameters of "Rescaling" layer. The input layer is followed by 3 hidden layers (Conv2D), with 16, 32, and 64 filters or feature maps outputted from each layer. In convolutional layers, 3 by 3 kernels are used. Padding is turned to ensure the feature map the same size as input picture. Activation in all three hidden layers is "relu". Each convolutional layer is followed by max. pooling layers with default hyperparameters. Then feature maps is flattened using Flatten function and dropout rate of 25% is set to avoid overfitting. The last, output layer has with 4 neurons representing 4 types of classification and activation type "softmax" which is used for multiclass classification.

Fifthly, we configure optimizer, loss and metrics using compile function and they were assigned to "adam", "sparse_categorical_crossentropy" and "accuracy" respectively.

Sixth, callbacks were assigned to monitor "validation_accuracy" parameter with patience value of 3, meaning the model with highest score of "validation_accuracy" will be saved as "best_model.keras" and no more training will happen after 3 trainings with no increase in value of "validation_accuracy" to prevent overfitting to dataset. Number of overall trainings was assigned to 20.

Seventh, the trained model saved in "best_model.keras" is then could be used to predict the contamination type based on image. To predict contamination type, we assign variable "img" with the name of image we are interested in and use "load_img" function to upload it to the same variable. Then "img_array" array is created from "img" variable with "img_to_array" function and extra axis is added to with "expand_dims" function, we then predict based with trained model giving us raw score saved in "predictions" variable which is then turned into probability with "softmax" function. As a result we receive a name of class with highest result and result itself.

Chapter 4

Results and Discussions

4.1 Discussions

4.1.1 Dielectric Dissipation Factor

Tables 4.1, 4.2, 4.3 show the results of the simulation. Table 4.1 contains the results of simulation at pollution profile of 2 mm. It could be clearly seen that iron as a pollutant strongly effect the DDF incresing its value almost 6 times, while aluminum and calcium effect less significant but they also cause alterations from clean insulator's DDF value. That is mainly caused by conductance value which is higher in case of conductors. We see gradual decrease of DDF with an decrease of conductance value. Decrease of DDF value basically means the decrease of energy loss in transmission system.

In Table 4.2 the DDF value for pollution profile of 4 mm is shown. In 4 mm profile, the increase of pollution mass and increase of pollution surface density is seen. Increase of mass and density effects on DDF value causing it to increase from value obtained at 2 mm pollution profile. Again in this case, the highest DDF value is reported with Iron pollution (3.189%) while aluminum and calcium also increased.

Table 4.3 contains the results of DDF simulation for pollution layer of 6 mm. As expected, the further increase of pollution layer will cause increase in DDF value. Thus the value of DDF with iron pollutant is 6.11%. The aluminum and calcium DDF cases show the values of 1.278 and 2.123. It should be noted that in all three pollution profiles the values of DDF for distilled water, soil, and concrete did not alter.

4.1.2 Image classification

In fitting process, 20 epochs(trainings) were planned, however in order to prevent overfitting of data, callbacks were initiated and result could be seen in Fig. A.6, where we see gradual increase of "validation_accuracy" throughout the fitting process with termination at epoch 12, meaning that the highest "validation_accuracy" value was obtained at 9th training with an value of 91.21%. Moreover the predict function was also tested on images devoted for prediction, they are not the same images used for training. The images named "IMG_0461.jpg", "IMG_0635.jpg", "IMG_0681.jpg", "IMG_0823.jpg", "IMG_1987.jpg", "IMG_0999.jpg", "IMG_1155.jpg", "IMG_1212.jpg". Results could be seen Figures 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8. Images used for prediction are attached in appendix B. Overall, predictions were done using images devoted for prediction, the results could be seen in Table 4.4. Our model is able to identify clean insulators and insulators contaminated with soil with 100% precision, while 1 mistake was made during the classification process of insulators contaminated with snow, the most difficult type of contamination to classify was water, 7 out of 10 predictions was correct, 2 times it was falsely predicted as clean and as contaminated with snow 1 time. Overall, the prediction accuracy was 90% which is promising

Table 4.1: Values at 2 mm of pollution

	Pollution mass (mg)	Pollution surface density ($g/cm\hat{2}$)	G (S)	C (pF)	tan
Clean	102	2.505	1.246 E-11	19.545	0.169
Water(distilled)	78836	1941.486	1.247 E-11	19.607	0.169
Soil	212858	5242.033	1.247 E-11	19.591	0.169
Aluminum	213646	5261.439	3.261 E-11	17.467	0.496
Iron	614924	15143.673	6.220 E-11	17.483	0.944
Calcium	121408	2989.903	2.061 E-11	17.599	0.311
Concrete	204974	5047.875	1.247 E-11	19.593	0.169

Table 4.2: Values at 4 mm of pollution

	Pollution mass (mg)	Pollution surface density ($g/cm\hat{2}$)	G (S)	C (pF)	tan
Water(distilled)	157672	3882.972	1.248 E-11	19.682	0.168
Soil	425716	10484.066	1.249 E-11	19.64	0.169
Aluminum	427292	10522.878	3.769 E-11	17.463	0.573
Iron	1229848	30287.346	2.115 E-10	17.595	3.189
Calcium	242816	11959.612	4.716 E-11	17.465	0.717
Concrete	409948	10095.75	1.247 E-11	19.666	0.168

Table 4.3: Values at 6 mm of pollution

	Pollution mass (mg)	Pollution surface density (g/cm$\hat{2}$)	G (S)	C (pF)	tan
Water(distilled)	236508	5824.458	1.251 E-11	19.81	0.168
Soil	638574	15726.099	1.250 E-11	19.686	0.169
Aluminum	640938	15784.317	8.417 E-11	17.477	1.278
Iron	1844772	45431.019	4.372 E-10	18.992	6.110
Calcium	364224	8969.709	1.401 E-10	17.513	2.123
Concrete	614922	15143.625	1.25 E-11	19.716	0.168

Table 4.4: Predicted and true classes

	True classes			
Predicted classes	Clean	Contaminated with snow	Contaminated with soil	Contaminated with water
Clean	10	0	0	0
Contaminated with snow	1	9	0	0
Contaminated with soil	0	0	10	0
Contaminated with water	2	1	0	7

```
img = "IMG_0461.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 267ms/step
This image most likely belongs to insulator_clean with a 35.81 percent confidence.
```

Figure 4.1: Clean polymer insulator

```
img = "IMG_0635.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 1s 560ms/step
This image most likely belongs to insulator_clean with a 46.67 percent confidence.
```

Figure 4.2: Clean glass insulator

```
img = "IMG_0681.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 66ms/step
This image most likely belongs to insulator_contaminated_snow with a 47.51 percent confidence.
```

Figure 4.3: Polymer insulator contaminated with snow

```
img = "IMG_0823.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 378ms/step
This image most likely belongs to insulator_contaminated_snow with a 47.41 percent confidence.
```

Figure 4.4: Glass insulator contaminated with snow

```
img = "IMG_1987.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 44ms/step
This image most likely belongs to insulator_contaminated_water with a 47.54 percent confidence.
```

Figure 4.5: Polymer insulator contaminated with water

```
img = "IMG_0999.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 44ms/step
This image most likely belongs to insulator_contaminated_water with a 46.06 percent confidence.
```

Figure 4.6: Glass insulator contaminated with water

```
img = "IMG_1155.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 74ms/step
This image most likely belongs to insulator_contaminated_soil with a 46.23 percent confidence.
```

Figure 4.7: Polymer insulator contaminated with soil

```python
img = "IMG_1212.jpg"
img = tf.keras.utils.load_img(img, target_size=(180, 180)
)
img_array = tf.keras.utils.img_to_array(img)
img_array = tf.expand_dims(img_array, 0)

predictions = best_model.predict(img_array)
score = tf.nn.softmax(predictions[0])

print(
    "This image most likely belongs to {} with a {:.2f} percent confidence."
    .format(class_names[np.argmax(score)], 100 * np.max(score))
)
```

```
1/1 [==============================] - 0s 50ms/step
This image most likely belongs to insulator_contaminated_soil with a 47.24 percent confidence.
```

Figure 4.8: Glass insulator contaminated with soil

Chapter 5

Conclusion

5.1 Dielectric Dissipation Factor

This research aimed to create different approach for frazzle determination of high-voltage insulators. One of the main parameters that show the efficiency of insulation is Dielectric Dissipation Factor. However, nowadays DDF is obtained from real-world experiments which might be inefficient. Thus the process was focused on using ANSYS Maxwell software as a basis for obtaining DDF via simulation. Using FEM, the insulator's surface was divided into small pieces with corresponding electric fields which they create. This allowed to calculate the whole electric field by which capacitance was found

To contaminate the surface of insulator, ANSYS Maxwell electrostatic solution type was used. The cap of insulator was at potential of 0V while the pin of insulator was at the potential of 10 kV which created the voltage of 10 kV. There was 3 nominal degrees of pollution, they are light, medium, and heavy. In addition, 3 types of pollutants were used they are water, soil, and aluminum which makes 10 simulations overall (extra simulation on clean insulator). According to the simulations, the tendency of DDF to increase with an increase of thickness of pollution layer was observed.

5.2 Image classifier

The second aim in this study was to create a robust and accurate model which will be able to automatically identify between different types of contamination of high voltage insulator, thus helping in aiding and in maintenance process of HV insulators. Through extensive training of a model, the capability of Convolutional Neural Network was showed. Results achieved in research clearly indicate the potential of Convolutional Neural Network and its future role in maintenance of power networks. Despite very promising results, this model requires further adjustments in terms of dataset variety and complexity of architecture. However, it is a big step towards real-time condition classification of high voltage insulators.

Bibliography

[1] Ahmed M. Ebid et al. "Ant Colony Optimization based algorithm to determine the optimum route for overhead power transmission lines". In: *Ain Shams Engineering Journal* 15.1 (2024), p. 102344. ISSN: 2090-4479. DOI: https://doi.org/10.1016/j.asej.2023.102344. URL: https://www.sciencedirect.com/science/article/pii/S2090447923002332.

[2] Yang Liu et al. "A New Noncontact Detection Method for Assessing the Aging State of Composite Insulators". In: *IEEE Transactions on Industrial Informatics* (2024), pp. 1–12. DOI: 10.1109/TII.2024.3352089.

[3] Ahmed Haiba and Ahmed Eliwa. "Artificial neural network analysis for classification of defected high voltage ceramic insulators". In: *Scientific Reports* 14 (Jan. 2024), pp. 1–14. DOI: 10.1038/s41598-024-51860-8.

[4] Mengxin Zhu et al. "Measuring the Dissipation Factor of Capacitive Equipment Under Damped AC Voltage". In: *IEEE Transactions on Instrumentation and Measurement* 73 (2024), pp. 1–12. DOI: 10.1109/TIM.2024.3350134.

[5] Shafi Jiskani et al. "Aerial identification of flashed over faulty insulator using binary image classification". In: *Mehran University Research Journal of Engineering and Technology* 43.1 (2024), pp. 225–233. ISSN: 2413-7219. DOI: 10.22581/muet1982.2401.3006. URL: https://publications.muet.edu.pk/index.php/muetrj/article/view/3006.

[6] J.L. Rasolonjanahary, L. Krahenbuhl, and A. Nicolas. "Computation of electric fields and potential on polluted insulators using a boundary element method". In: *IEEE Transactions on Magnetics* 28.2 (1992), pp. 1473–1476. DOI: 10.1109/20.123974.

[7] Rong Zeng et al. "Measurement of electric field distribution along composite insulators by integrated optical electric field sensor". In: *IEEE Transactions on Dielectrics and Electrical Insulation* 15.1 (2008), pp. 302–310. DOI: 10.1109/TDEI.2008.4446764.

[8] N. A. Othman et al. "Simulation of voltage and electric-field distribution for contaminated glass insulator". In: (2013), pp. 116–120. DOI: 10.1109/SCOReD.2013.7002554.

[9] Ali Ahmed Salem et al. "Investigation of High Voltage Polymeric Insulators Performance under Wet Pollution". In: *Polymers* 14.6 (2022). ISSN: 2073-4360. DOI: 10.3390/polym14061236. URL: https://www.mdpi.com/2073-4360/14/6/1236.

[10] Ebrahim Akbari et al. "Finite Element Analysis of Disc Insulator Type and Corona Ring Effect on Electric Field Distribution over 230-kV Insulator Strings". In: *International Journal of Engineering & Technology* 1 (Aug. 2012), pp. 407–419. DOI: 10.14419/ijet.v1i4.330.

[11] Farah Adilah Jamaludin et al. "Effect of RTV Coating Material on Electric Field Distribution and Voltage Profiles on Polymer Insulator under Lightning Impulse". In: (2018), pp. 1–6. DOI: 10.1109/ICLP.2018.8503296.

[12] Arshad et al. "Effect of pollution severity on electric field distribution along a polymeric insulator". In: *2015 IEEE 11th International Conference on the Properties and Applications of Dielectric Materials (ICPADM)* (2015), pp. 612–615. URL: https://api.semanticscholar.org/CorpusID:13124911.

[13] Ali Ahmed Salem et al. "Pollution Flashover Characteristics of Coated Insulators under Different Profiles of Coating Damage". In: *Coatings* 11.10 (2021). ISSN: 2079-6412. DOI: 10.3390/coatings11101194. URL: https://www.mdpi.com/2079-6412/11/10/1194.

[14] C. S. Ilomuanya, A. Nekahi, and S. Farokhi. "Acid Rain Pollution Effect on the Electric Field Distribution of a Glass Insulator". In: (2018), pp. 1–4. DOI: 10.1109/ICHVE.2018.8642231.

[15] Dongdong Zhang et al. "Simulation Study on the Effects of DC Electric Field on Insulator Surface Pollution Deposit". In: *Energies* 11.3 (2018). ISSN: 1996-1073. DOI: 10.3390/en11030626. URL: https://www.mdpi.com/1996-1073/11/3/626.

[16] Waseem Rawat and Zenghui Wang. "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review". In: *Neural Computation* 29.9 (2017), pp. 2352–2449. DOI: 10.1162/neco_a_00990.

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[18] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25 (2012).

[20] Christian Szegedy et al. "Going deeper with convolutions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594.

[21] Taranjit Kaur and Tapan Kumar Gandhi. "Automated Brain Image Classification Based on VGG-16 and Transfer Learning". In: *2019 International Conference on Information Technology (ICIT)*. 2019, pp. 94–98. DOI: 10.1109/ICIT48102.2019.00023.

[22] T. Stolarski, Y. Nakasone, and S. Yoshimoto. *Engineering Analysis with ANSYS Software*. ELSEVIER, 2006.

[23] A. Martyanov and N. Neustroyev. *ANALYSIS OF ELECTROMECHANICAL SYSTEMS USING ANSYS MAXWELL*. Alternative Energy and Ecology, 2014.

[24] Nouruddeen Bashir, Hussein Ahmad, and Mohd Shaffuan Suddin. "Ageing studies on transmission line glass insulators using dielectric dissipation factor test". In: (2010), pp. 1062–1066. DOI: 10.1109/IPECON.2010.5696973.

[25] Issouf Fofana and Yazid Hadjadj. "Electrical-Based Diagnostic Techniques for Assessing Insulation Condition in Aged Transformers". In: *Energies* 9.9 (2016). ISSN: 1996-1073. DOI: 10.3390/en9090679. URL: https://www.mdpi.com/1996-1073/9/9/679.

Appendix A

Code

```
import numpy as np
import tensorflow as tf

from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.models import Sequential
```

Figure A.1: Library installation and import

```
data_directory="ML2#2/ML2#2/dataset"
```

```
batch_size = 32
img_size = (180, 180)
```

Figure A.2: Batch size, dataset library and image size assignment

```
training_set = tf.keras.utils.image_dataset_from_directory(
  data_directory,
  validation_split=0.2,
  subset="training",
  seed=123,
  image_size=img_size,
  batch_size=batch_size)
```

```
Found 2275 files belonging to 4 classes.
Using 1820 files for training.
```

```
validation_set = tf.keras.utils.image_dataset_from_directory(
  data_directory,
  validation_split=0.2,
  subset="validation",
  seed=123,
  image_size=img_size,
  batch_size=batch_size)
```

```
Found 2275 files belonging to 4 classes.
Using 455 files for validation.
```

Figure A.3: Division of dataset

```
model = Sequential([
  layers.Rescaling(1./255, input_shape=(180, 180, 3)),
  layers.Conv2D(16, 3, padding='same', activation='relu'),
  layers.MaxPooling2D(),
  layers.Conv2D(32, 3, padding='same', activation='relu'),
  layers.MaxPooling2D(),
  layers.Conv2D(64, 3, padding='same', activation='relu'),
  layers.MaxPooling2D(),
  layers.Flatten(),
  layers.Dropout(0.25),
  layers.Dense(128, activation='relu'),
  layers.Dropout(0.25),
  layers.Dense(4, activation = 'softmax')
])
```

Figure A.4: Model architecture

```
class_names = training_set.class_names
```

```
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
```

Figure A.5: Compilation

```
filters_options_layer1 = [16, 32, 64]  # Options for number of filters for the first Layer
filters_options_layer2 = [16, 32, 64]  # Options for number of filters for the second Layer
filters_options_layer3 = [16, 32, 64]  # Options for number of filters for the third Layer
kernel_sizes_options = [(3, 3), (5, 5), (7, 7)]  # Kernel sizes options
for filters1 in filters_options_layer1:
    for filters2 in filters_options_layer2:
        for filters3 in filters_options_layer3:
            for kernel_size in kernel_sizes_options:
                print(f"Training model with filters [{filters1}, {filters2}, {filters3}] and kernel size {kernel_size} for all layers")
                model = Sequential([
                    layers.Rescaling(1./255, input_shape=(180, 180, 3)),
                    layers.Conv2D(filters=filters1, kernel_size=kernel_size, padding='same', activation='relu'),
                    layers.MaxPooling2D(),
                    layers.Conv2D(filters=filters2, kernel_size=kernel_size, padding='same', activation='relu'),
                    layers.MaxPooling2D(),
                    layers.Conv2D(filters=filters3, kernel_size=kernel_size, padding='same', activation='relu'),
                    layers.MaxPooling2D(),
                    layers.Flatten(),
                    layers.Dropout(0.25),
                    layers.Dense(128, activation='relu'),
                    layers.Dropout(0.25),
                    layers.Dense(4, activation = 'softmax')
                ])
                model.compile(optimizer='adam',
                              loss='sparse_categorical_crossentropy',
                              metrics=['accuracy'])
                my_callbacks = [
                    keras.callbacks.EarlyStopping(
                        monitor="val_accuracy",
                        patience=3),
                    keras.callbacks.ModelCheckpoint(
                        filepath="best_model.keras",
                        monitor="val_accuracy",
                        save_best_only=True,
                        verbose=1)
                ]
                history = model.fit(training_set,
                      batch_size = 32,
                      epochs = 20,
                      validation_data = validation_set,
                      callbacks=my_callbacks)
best_model = keras.models.load_model("best_model.keras")
```

Figure A.6: Architecture of model with nested loops

Appendix B

Prediction images

(a) IMG_0461

(b) IMG_0635

Figure B.1: Clean insulators

(a) IMG_0681

(b) IMG_0823

Figure B.2: Insulators contaminated with snow

(a) IMG_1987

(b) IMG_0999

Figure B.3: Insulators contaminated with water

(a) IMG_1987 **(b)** IMG_0999

Figure B.4: Insulators contaminated with soil