SKIN CANCER DETECTION AND CLASSIFICATION USING MACHINE LEARNING TECHNIQUES

Uddhav Vaze Sanjivani Palekar Aishwarya Galagali

CERTIFICATE

This is to certify that the project entitled

"Skin Cancer Detection and Classification Using Machine Learning Techniques"

is a record of work done by

Ms. Aishwarya G. Galagali
Ms. Sanjivani D. Palekar
&
Mr. Uddhav S. Vaze

of M.SC. Part II (Electronics) 2021-2022

The candidates themselves have worked on the project during the period of study under my guidance and to the best of my knowledge, it has not previously formed the basis of award of any previous degree or diploma at Goa University or elsewhere.

Program Director (Electronics)	**Examiner**	**Project Guide**

"SKIN CANCER DETECTION AND CLASSIFICATION USING MACHINE LEARNING TECHNIQUES"

By

Ms. Aishwarya G. Galagali
Ms. Sanjivani D. Palekar
&
Mr. Uddhav S. Vaze

M.Sc. Part II Electronics

School of physical and applied sciences

Goa University

2021-2022

ACKNOWLEDGEMENT

Firstly, we want to thank the almighty for the grace and blessing bestowed upon us in completing our thesis. We have given our best in this project.

However, it would have not been possible without the support and help of many individuals.

We express our deep gratitude to all the teaching faculty Dr. R. S. Gad, Dr. J. S. Parab, Dr. Marlon Sequeira, Dr. Narayan Vetrekar, and Dr. Aniketh Gaonkar for their encouragement, assistance, and guidance in the project.

We would also thank Ph. D colleagues and office staff for their kind support and sharing their bit of knowledge on the way without which many things wouldn't be possible.

We would also like to express our gratitude towards our parents for their kind co-operation and encouragement.

Our thanks and appreciations also go to the people who have willingly helped us out with their abilities

Last but not least a big thank you to the contributors on the internet for providing references that were used for the report.

DECLARATION

We the students of the "Goa University" batch of M.Sc. Electronics hereby solemnly declare that this project under the title "Skin Cancer Detection and Classification Using Machine Learning Techniques" is a record of work that had been composed by us, and this report has not been submitted anywhere else for the award of any diploma or degree to the best of our knowledge.

Aishwarya G. Galagali **Sanjivani D. Palekar** **Uddhav S. Vaze**

TABLE OF CONTENTS

ABSTRACT

Skin cancer is one of the most severe diseases, and medical imaging is among the main tools for cancer diagnosis. The images provide information on the evolutionary stage, size, and location of tumor lesions. This paper focuses on classifying skin lesion images to analyze the performance of Convolutional Neural Networks (CNNs) in distinguishing different skin lesions. Three pre-trained CNN models: MobileNet, ResNet-50, and VGG-16 based on transfer learning are compared using the HAM10000 dataset and ISIC 2019 dataset, taking advantage of ImageNet weights. The results obtained by the three models demonstrate accuracies of 92.42%, 71.60%, and 87.70%, respectively for the HAM10000 dataset, and accuracies of 93.63%, 61%, and 75.24% respectively for the ISIC 2019 dataset. Finally, the best model is tested on the ISIC 2019 dataset showing an accuracy of 93.63%.

We proposed a model and got an accuracy of 97.04% and 93.52% for the HAM10000 and ISIC 2019 datasets respectively. The proposed methodology using CNN represents a helpful tool to diagnose skin cancer disease accurately.

CHAPTER I

1.1 INTRODUCTION

Cancer is a generic term for a large group of diseases that can affect any part of the body. Other terms used are malignant tumors and neoplasm. One defining feature of cancer is the rapid creation of abnormal cells that grow beyond their usual boundaries, and which can then invade adjoining parts of the body and spread to other organs; the latter process is referred to as metastasis. Widespread metastases are the primary cause of death from cancer.

Cancer is a leading cause of death worldwide, accounting for nearly 10 million deaths in 2020. The most common in 2020 (in terms of new cases of cancer) were:

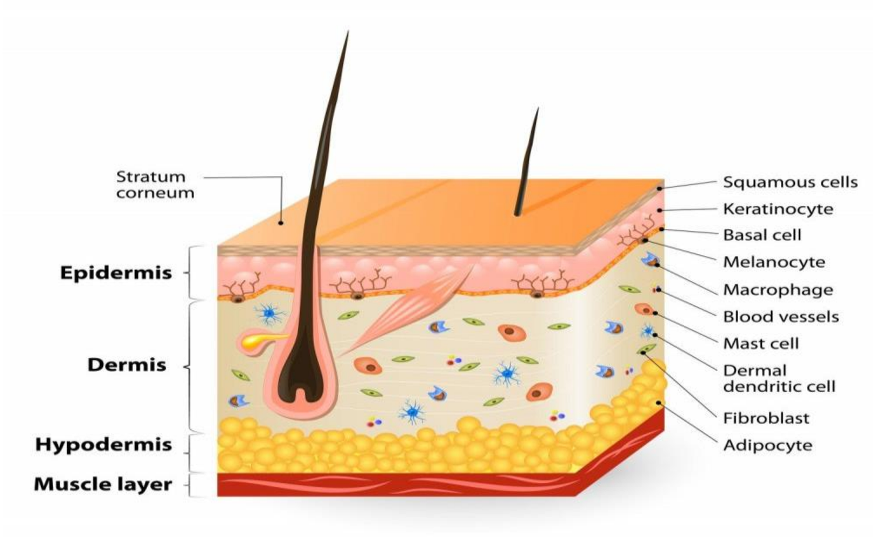
- breast (2.26 million cases);
- lung (2.21 million cases);
- colon and rectum (1.93 million cases);
- prostate (1.41 million cases);
- skin (non-melanoma) (1.20 million cases); and
- Stomach (1.09 million cases).

Each year, approximately 400,000 children develop cancer.

Cancer arises from the transformation of normal cells into tumor cells in a multi-stage process that generally progresses from a pre-cancerous lesion to a malignant tumor. These changes are the result of the interaction between a person's genetic factors and three categories of external agents, including:

- physical carcinogens, such as ultraviolet and ionizing radiation;
- chemical carcinogens, such as asbestos, components of tobacco smoke, alcohol, aflatoxin (a food contaminant), and arsenic (a drinking water contaminant); and
- Biological carcinogens, such as infections from certain viruses, bacteria, or parasites.

- WHO, through its cancer research agency, the International Agency for Research on Cancer (IARC), maintains a classification of cancer-causing agents. The incidence of cancer rises dramatically with age, most likely due to a build-up of risks for specific cancers that increase with age. The overall risk accumulation is combined with the tendency for cellular repair mechanisms to be less effective as a person grows older. [22]
- The largest organ in the human body is the skin. The disorganized and uncontrolled growth of skin cells leads to skin cancer formation and cancer can rapidly grow to other body parts. Skin Cancer is an emerging global health problem considering the increasing prevalence of harmful ultraviolet rays in the earth's environment [9]. Skin cancer is a dangerous and widespread disease. [1] Each year there are approximately 5.4 million new cases of skin cancer are recorded in the USA alone. The global statistics are equally alarming. [19] Skin cancer is one of the significant contributors to the cause of death over the world. [18]
- Skin Cancer initially occurs on the upper layer of the skin, the epidermis, where it is noticeable and can be seen by human eyes [18] the epidermis is the superficial layer of skin and mainly consists of three cells: Squamous cells, Basal cells, and Melanocytes. The outermost cells are Squamous and the lowermost layer cells are Basal cells of the epidermis. Melanocytes protect deeper layers of skin from exposure of Sun by producing melanin, a brown pigment substance. When these cells experience excessive ultraviolet light exposure, the DNA mutations induced affect the growth of skin cells and eventually shape into skin cancer. [7]



[21]

Figure 1

There are three major types of skin cancer — basal cell carcinoma, squamous cell carcinoma, and melanoma. Skin cancer develops primarily on areas of sun-exposed skin, including the scalp, face, lips, ears, neck, chest, arms, hands, and the legs in women. But it can also form on areas that rarely see the light of day — your palms, beneath your fingernails or toenails, and your genital area. Skin cancer affects people of all skin tones, including those with darker complexions. When melanoma occurs in people with dark skin tones, it's more likely to occur in areas not normally exposed to the sun, such as the palms of the hands and soles of the feet. [24]

Table 1

Sr. No.	Cancer Type
1	BKL
2	NV
3	DF
4	MEL
5	VASC
6	BCC
7	AKIEC
8	SCC

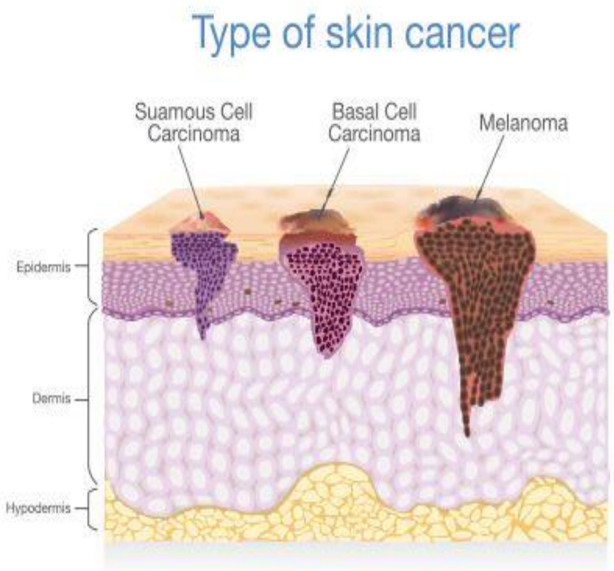


Figure 2

The deadliest form of skin cancer is melanoma, and its prevalence has been rapidly rising in the last 30 years. Melanoma of the skin is the 17th most common cancer worldwide. It is the 13th most common cancer in men and the 15th most common cancer in women. There were more than 150,000 new cases of melanoma of the skin in 2020. Non-melanoma skin cancer is often excluded from the reporting of cancer statistics. [20] Early Diagnosis can increase one's chances of survival. [20]

Dermatologists established the classification system for an accurate diagnosis of skin disorders. Because there are diverse characteristics of skin cancer, similarity to benign lesions and specific lesions seen from diseases, distinguishing skin cancer from other skin disorders is important in dermatology clinics. Although skin cancer can be early detected by direct interference, such visual similarity of lesions and various patterns make it difficult to diagnose the exact type of cancer. [28] The results are complex, time-consuming, subjective, and error-prone. These approaches depend on the dermatologist's expertise.
The cause of this fact is the complex nature of the skin lesion images due to two factors:

i) The characteristics of the lesions include texture, size, color, shape, and location.

ii) The presence of multiple artifacts in the images, such as hair, veins, and charts of color calibration and ruler marks [3].

Over the several years, dermoscopy has been enabled to generate a uniquely large number of detailed images of skin cancer and dermoscopy-based non-invasive diagnosis has been developed rapidly. [28]

Recently, as a natural next step, computers have learned the features that optimally represent image characteristics, leading to the development of a new machine learning branch called deep learning. The deep learning method can automatically mine the deep-seated nonlinear relationship in target images and does not need to establish feature estimation and extraction that are required in the traditional image recognition methods. The first kind of deep learning model used for skin lesion image processing was the convolutional neural network (CNN). This architecture was demonstrated to exceed a dermatologist's performance in distinguishing melanoma from non-melanoma [3] Li et al. [30] provided an extensive review of CNN deep learning models and compared the most popular architectures such as AlexNet, VGG, GoogleNet, Inception, ResNet, DenseNet, and others. The most accurate model is based on residual learning and separable convolution, with an accuracy of 99.5% [21]. However, this accuracy was achieved for a binary problem.

When the skin lesion classification problem is treated as a multi-class problem; the CNN models require additional steps, either data augmentation or an ensemble of classifiers to reach accuracy above 80%. Another deep learning technique named transfer learning improves the performance by taking advantage of previously trained architectures. [3]

Transfer learning turns out to be useful when dealing with relatively small datasets, e.g. medical images, which are harder to obtain in large numbers than other datasets. Instead of training a neural network from scratch, which would require a significant amount of data and a high computational time, it is often convenient to use a pre-trained model and just fine-tune its performance to simplify and speed up the process. [6]

Several studies investigated different machine learning techniques for the diagnosis of different types of cancer. Most of these studies employed classifiers with relatively simple structures and trained on a group of hand-crafted features extracted from the images. Most machine learning techniques require high computational time for accurate diagnosis and their performance depends on the selected features that characterize the cancerous region. Deep learning techniques and Convolutional Neural Networks (CNNs) have become an important approach for automated diagnosis of different types of cancer. [1] Although the method is complicated, deep learning algorithms have shown exceptional performance in visual tasks and even outperformed humans in gaming. [9]

The convolutional neural network (CNN), one of the most powerful deep learning techniques proved to be superior to traditional algorithms. These networks provide the flexibility of extracting discriminatory features from images that preserve the spatial structure and could be developed for region recognition and medical image classification. [29]

Over the last decades, the research on computer-aided diagnostic (CAD) techniques has intensified to verify, support, or provide a second opinion to physician's decisions (4). With the current resurgence of interest in machine learning, deep learning, and neural networks, automated skin cancer classification has been an important topic of research. The success of neural networks in the ImageNet Large Scale Visual Recognition Challenge fostered various deep learning-based solutions for multiclass skin lesion classification (5).

The CAD system generally consists of four main steps: image pre-processing, segmentation, feature extraction, and classification. Most machine learning techniques require high computational time for accurate diagnosis and their performance depends on the selected features that characterize the cancerous region. Deep learning techniques and Convolutional Neural Networks (CNNs) have become an important approach for automated diagnosis of different types of cancer. Deep learning has achieved impressive results in image classification, including skin lesion analysis. In image classification tasks, transfer learning and data augmentation are employed to overcome the lack of data and to reduce the computational and memory requirements. [6]

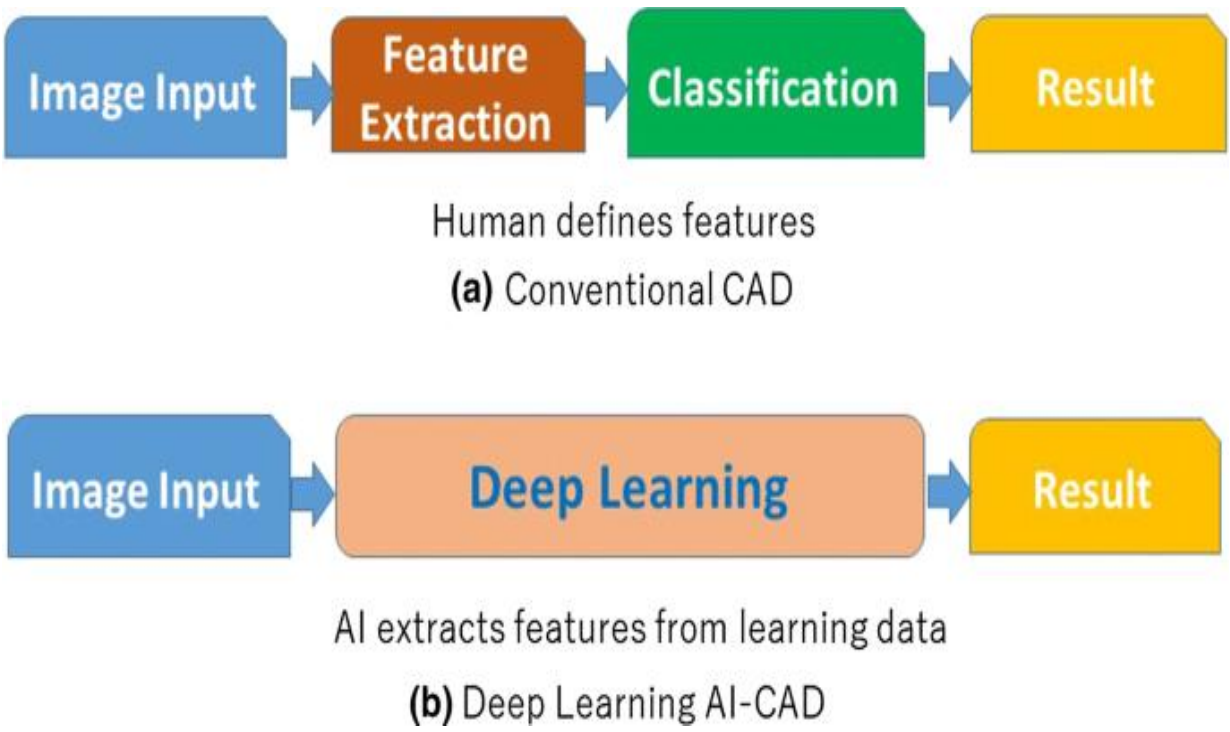


Figure 3

Machine Learning is one of the fastest-growing fields which have witnessed exponential growth in the technical world. There is no best language for machine learning it depends on what you want to build. To work in this field, you just need to learn only one particular programming language very well based on your comfort, project requirements, and predilections.

Python Programming Language

Python is an object-oriented programming language created by Guido Rossum in 1989. It's ideally designed for rapid prototyping of complex applications. It has interfaces to several operating system calls and libraries and is extensible to C or C++. Many large companies use the Python programming language including NASA, Google, YouTube, BitTorrent, etc. Python is widely employed in computer science, Artificial Intelligence, Natural Language Generation, Neural Networks, and other advanced fields of applied Science. [66]

Python is one of those rare languages which may claim to be both simple and powerful.

Different platforms for machine learning

Machine learning platforms provide users with the tools necessary to develop, deploy, and improve machine learning — specifically, machine learning algorithms. Machine learning platforms automate data workflows, accelerate data processing, and optimize related functionality.

Google Colab

Google Colab is a great platform for deep learning enthusiasts, and it also can be used to test basic machine learning models, gain experience, and develop an intuition about deep learning aspects like hyperparameter tuning, preprocessing data, model complexity, over-fitting, and more.

JupyterLab

JupyterLab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality.

1.2 OBJECTIVE

The incidence rate of skin cancers is increasing worldwide annually. Using machine learning and deep learning for skin lesion classification is one of the essential research topics. This paper focuses on classifying skin lesion images from publically available datasets: the HAM10000 dataset and ISIC 2019 dataset to analyze the classification performance of Convolutional Neural Networks (CNNs). We studied three pre-trained models namely, Mobile-Net, ResNet-50, and VGG16 using Transfer Learning. Also, we proposed a new model, and a comparative study between those models is done.

1.3 LITERATURE SURVEY

Recent studies have reported excessive exposure to ultraviolet rays as a major factor in developing skin cancer. The most effective solution to control the death rate for skin cancer is a timely diagnosis of skin lesions as the five-year survival rate for melanoma patients is 99 percent when diagnosed and screened at the early stage. Considering the inability of dermatologists to accurate diagnosis of skin cancer, there is a need to develop an automated efficient system for the diagnosis of skin cancer.

In recent years, several classification approaches have been proposed for the automatic detection of skin cancer from dermatoscopic images. CNN approaches have completely dominated the skin lesion classification process and this related work will be just a drop in the bucket.

The proposed approaches are evaluated using the HAM10000 dataset taken from the ISIC 2018 challenge, consisting of 10,015 dermatoscopic images belonging to 7 classes, and the ISIC 2019 dataset taken from the ISIC 2019 challenge, consisting of 25,331 dermatoscopic images belonging to 8 classes.

Some paper focuses on the importance of the Python programming language. It has interfaces to many operating system calls and libraries and is extensible to C or C++. Many large companies that use the Python programming language include NASA, Google, YouTube, BitTorrent, etc. Python is widely used in Artificial Intelligence, Natural Language Generation, Neural Networks, and other advanced fields of Computer Science. [66]

1. Priti Bansal (2022)

Three methods are proposed to remove hair from dermatoscopic images by applying various morphological operations and integration of features extracted using both HC techniques and DLMs namely ResNet50V2 and EfficientNet-B0 are proposed to enhance the classification accuracy of melanoma detection from HAM10000 and PH2 datasets. [1]

2. Long Hoang (2022)

A novel method is proposed to segment the skin image using the entropy-based weighting (EW) and first-order cumulative moment (FCM) of the skin image. A two-dimensional wide-ShuffleNet network is applied to classify the segmented image after applying EWFCM. Based on numerical results on HAM10000 and ISIC2019 datasets, the proposed framework is more efficient and accurate than state-of-the-art methods. [2]

3. Juan Pablo Villa-Pulgari (2021)

An optimization process of transfer learning models is proposed for multiple skin lesion classification, successfully addressed by using DenseNet-201, Inception-V3, and Inception-ResNet-V2 architectures and pre-training the weights of each model with ImageNet. Model performance was compared using two datasets HAM10000 and ISIC 2019. Since the datasets have a class imbalance, four experiments are conducted: pre-trained models without data augmentation, without optimization, with data augmentation, and the proposed optimization. As a result, the convolutional layers added to the former transfer model improved the overall performance. [3]

4. Camilo Calderon (2021)

Skin lesion images are classified through a deep learning approach with a non-traditional bilinear CNN architecture. The model is trained in a transfer learning and fine-tuning way. The proposed method called BILSK is tested on the HAM10000 dataset. This approach gets the highest accuracy against state-of-the-art techniques in the classification of current skin lesions dataset. [4]

5. Karar Ali (2021)

The classification performance of the EfficientNets B0-B7 is investigated on the HAM10000 dataset of dermatoscopic images. ImageNet pre-trained weights are used to perform transfer learning and fine-tune the CNNs for the HAM10000 dataset. Precision, Recall, Accuracy, F1 Score, Specificity, Roc Auc Score, and Confusion Matrices evaluated the EfficientNets B0-B7 performance on this imbalanced multiclass classification task. This paper also presents the per-class classification exactitudes in the form of Confusion Matrices for all eight models. [5]

6. Wessam M. Salamaa (2021)

Two deep neural networks ResNet50 and VGG16 with different strategies, with and without preprocessing and with and without SVM, are used. A fully connected layer is replaced with SVM in both previous networks to achieve high rates in the skin cancer classification problem. Preprocessing steps are employed to improve dataset quality and therefore increase framework performance. Three data augmentation and transfer learning techniques are employed on the most common dermatoscopic image databases in the field ISIC (2017), HAM10000 (2018), and ISBI (2016). [6]

7. Saket S. Chaturvedi, Jitendra V. Tembhurne (2020)

An automated computer-aided diagnostic system for MCS cancer classification with exceptionally high accuracy is proposed. The proposed method outperformed both expert dermatologists and previously proposed deep learning methods for MCS skin cancer classification. A comparative study is conducted to analyze the performance of five pre-trained convolutional neural networks and four ensemble models to determine the best method for skin cancer classification on the HAM10000 dataset. Extensive research performed in determining the best set-up of hyper-parameters for five models pre-trained on ImageNet namely Xception, InceptionV3, InceptionResNetV2, NASNetLarge and ResNetXt101 and their ensembles InceptionV3 + Xception, InceptionResNetV2 + Xception, InceptionResNetV2 + ResNetXt101, and InceptionResNetV2 + ResNetXt101 + Xception. These models are fine-tuned further on the HAM10000 dataset using Transfer Learning to learn domain-specific features of skin cancers. [7]

8. Karl Thurnhofer-Hemsi (2020)

An end-to-end system is proposed to classify skin diseases with the use of deep neural networks, which will be already trained for its use by the user without requiring any parameter tuning. This paper evaluates the performance of the state-of-art pre-trained deep networks for melanoma detection by applying transfer learning using the HAM10000 dataset. To generate an efficient classification model by using few images and dealing with unbalanced classes, for this purpose a hierarchical classifier based on two levels (and two neural models) is proposed, where the first neural network at the first level separates the nevi class from the rest, and the second one classifies the other six classes. [8]

9. Saket S. Chaturvedi, Kajol Gupta (2020)

This study explores an efficient automated method for the classification of dermoscopy skin cancer images. A MobileNet convolutional neural network pre-trained on approximately 12,80,000 images from the 2014 ImageNet Challenge is utilized and fine-tuned on the HAM10000 dataset employing transfer learning. The MobileNet model classified skin lesion images with performance better or comparable to expert dermatologists for seven classes. Data analysis was conducted on the dermoscopy images of skin cancer to uncover the relationship of skin cancer with several parameters to strengthen the understanding of skin cancer. [9] He achieved an accuracy of 83.1% for MobileNet.

10. Duyen N.T. Le (2020)

A deep convolutional neural network (CNN) is proposed with high accuracy for classifying skin lesions into seven categories. Transfer learning technique is applied by employing pre-trained ResNet50, VGG16, and MobileNet models in combination with focal loss and class weighted. CNN-based deep learning provides an end-to-end approach, which achieves high accuracy without the need for feature engineering. [10]

Accuracy for ResNet50: 93%

11. José Ariel Camacho-Gutiérrez (2022)

Two fractal signatures are proposed, named Statistical Prism-Based Fractal Signature and Statistical Fractal Signature. The performance of ensemble models is analyzed to identify the best classifier space using the proposed fractal signatures. Furthermore, the impact on the results of the computer-assisted diagnosis model is analyzed when databases with unbalanced classes are used. The importance of reporting the performance metrics of the training and testing parts is studied to ensure that the models are reproducible and the need to repeat the experiments according to the central limit theorem to ensure that the models are not biased. [11]

12. Di Zhuang (2022)

A cost-sensitive multi-classifier active fusion framework for skin lesion classification is proposed, where two types of weights are defined: the objective weights and the subjective weights. The objective weights are designed according to the reliability of the classifier to recognize the particular skin lesions, which are determined by the prior knowledge obtained through the training phase.

The subjective weights are designed according to the relative confidence of the classifiers while recognizing a specific previously ‘‘unseen” sample which is calculated by the posterior knowledge obtained through the testing phase. While designing the objective weights, a customizable cost matrix is utilized to enable the ‘‘cost-sensitive” feature infusion framework, were given a sample, different outputs of a classifier should result in different costs. In the experimental evaluation, 96 base classifiers are trained as the input of the fusion framework, utilizing twelve CNN architectures on the ISIC Challenge 2019 research dataset for skin image analysis.

Two static fusion baseline approaches (i.e., max voting and average fusion) and two state-of-the-art active fusion approaches (i.e., MCE-DW [5] and DES-MI [6]) are compared with the CS-AF framework. Experimental results show that the CS-AF framework consistently outperforms the static fusion baseline approaches and the state-of-the-art competitors in terms of accuracy, and always achieves lower total cost. [12]

13. Benny Wei-Yun Hsua (2022)

A novel loss function with hierarchical information learning is proposed and a late fusion framework is presented in terms of contrastive learning for skin lesion classification. Additionally, the experiments with k-fold cross-validation are conducted to ensure the robustness of the evaluation for our proposed methods under different data distributions. [13]

14. Yongwei Wanga (2022)

A novel knowledge distillation framework is developed, termed SSD-KD that integrates diverse conventional knowledge with a novel type of intra-instance relational knowledge to improve the auto-diagnosis of multiple types of skin diseases based on lightweight deep learning models.

Dual relational knowledge distillation architecture is formulated by introducing different relational representations and softened network outputs to distill and transfer diverse knowledge from the teacher model to the student model. Moreover, the self-supervised knowledge distillation strategy is incorporated into the framework to enable the student model to capture richer structured knowledge from the self-supervision predictions of the teacher model.

Additionally, to make the proposed SSD-KD framework more suitable for the skin lesions auto-detection task, which always suffers from data imbalance problems, the general cross-entropy loss function is replaced in conventional KDs with an improved weighted version to benefit the skin disease categories with fewer subjects. The proposed method has been evaluated in the publicly available skin lesion datasets ISIC 2019. The comparison with state-of-the-art methods and ablation studies are also performed to demonstrate the effectiveness of the proposed framework. [14]

15. SamiaBenyahia (2021)

This paper tries to acquire the features of dermoscopic images with a pre-trained CNN model and inject the obtained characteristics into several classifiers. Extracted features are classified using various algorithms: k-nearest neighbor (KNN), decision tree, Linear Discriminant, Naive Bayes, hybrid kernel-based SVM, and finally ensemble classifiers. 17 commonly pre-trained CNN architectures are used as feature extractors and 24 machine learning classifiers to evaluate the classification of skin lesions from two different datasets ISIC 2019 and PH2. [15]

16. Max Torop (2021)

The performance of unsupervised OOD detection methods is illustrated in the ISIC 2019 challenge dataset. The potential for unsupervised OOD detection in dermoscopy images is indicated in the results. Evidence for the difficulty gap between the HAM and BCN sources in ISIC 2019 was found. [16]

17. Zillur Rahman (2021)

A deep neural network-based system for skin lesion classification is proposed exploiting five state-of-the-art architectures, ResNet, DenseNet, Xception, ResNeXt, and SeResNeXt. The HAM10000 and ISIC dataset are used. The high data imbalance problem was solved by using cost-sensitive learning. Since the images in the datasets are affected by the hair and other noises, they were removed using the image in painting technique and resized to match the input size of the training models. The number of training images, as well as the variations among them, was increased by the data augmentation method.

The models were trained with various hyper-parameter combinations to find the best result. An average ensemble was designed by combining all the base models which shows a significant increment in classification results and a weighted average ensemble model where the best weights combination was found by the grid search method. [17]

18. MOHAMED A. KASSEM (2020)

The architecture of GoogleNet is modified by adding more filters to each layer, to enhance features and reduce noise. The last three layers in GoogleNet are replaced in two different ways:

a. The last three layers have been dropped out and replaced with new fully connected, SoftMax, and classification output layers. By using SoftMax, the proposed model works through binary and multi-class.

b. The second way was done by dropping out the last two layers only and keeping the original fully connected layer of GoogleNet as feature extractors.

A bootstrap multi-class support vector machine is used to classify images. The change aims to detect unknown images (outliers).

In the proposed model, no pre-processing such as noise reduction, segmentation, or enhancement is used. The weights of All layers are fine-tuned also, the proposed model does not over-fit even with some classes containing a small number of images. Dataset is classified with high-performance measures in two different ways. [18]

19. Bill Cassidy (2022)

The main contributions of this paper are as follows:

1. We analyze the usage of ISIC image datasets with a selection of well-cited research papers from the past 3–4 years and identify related issues.

2. We propose a duplicate removal strategy to curate the datasets. By removing the duplicate images (overlap images between and within the test and training sets), we produced a cleaned (non-duplicate) dataset and a balanced dataset.

3. We benchmark the curated balanced training set using 19 state-of-the-art deep learning architectures for melanoma recognition. We evaluate the performance of our benchmark algorithms on the ISIC 2020 testing set (on Kaggle) for binary classification (melanoma and non-melanoma), with additional analysis of the ISIC 2017 testing set.

4. We provide recommendations for future research and share our research findings on our GitHub repository. [37]

20. GuissousAllaEddine (2022)

This paper reports the methods and techniques we have developed for classifying dermoscopic images (task 1) of the ISIC 2019 challenge dataset for skin lesion classification, our approach aims to use an ensemble deep neural network with some powerful techniques to deal with unbalanced data sets as its the main problem for this challenge in a move to increase the performance of CNNs model. [38]

21. MehwishDildar (2021)

This paper presents a detailed systematic review of deep learning techniques for the early detection of skin cancer. Research papers published in well-reputed journals, relevant to the topic of skin cancer diagnosis, were analyzed. Research findings are presented in tools, graphs, tables, techniques, and frameworks for better understanding. [39]

22. Md. AminurRab Rahul (2020)

The author chose transfer learning with four popular architectures: VGG16, VGG19, MobileNet, and InceptionV3.
They used a confusion matrix to measure the performance.
Accuracy
VGG16: 87.42%
VGG19: 85.02%
MobileNet: 88.22%
InceptionV3: 89.81% [62]

23. Md Shahin Ali (2021)

They proposed a deep convolutional neural network (DCNN) model based on a deep learning approach for the accurate classification of benign and malignant skin lesions. DCNN model is compared with transfer learning models such as AlexNet, ResNet, VGG-16, DenseNet, MobileNet, etc.
Accuracy
AlexNet: 88.81%
ResNet: 85.20%
VGG16: 86.09%
DenseNet: 85.25%
MobileNet: 82.63%
Proposed model: 90.16% [63]

24. Maximilliano Lucius (2020)

This study evaluated whether deep learning frameworks trained in large datasets can help non-dermatologist physicians improve their accuracy in categorizing the seven most common pigmented skin lesions.
Different deep neural networks (DNNs) (n = 8) were trained based on a random dataset constituted of 8015 images.
Accuracy
ResNet34: 75.32%
ResNet50: 74.56%
ResNet101: 75.62%
SEResNet50: 77.82%
VGG16: 76.85%
VGG19: 74.21%
EfficientNetB5: 74.05%
MobileNet: 82.47%

25. Aryan Mobiny (2019)

They investigated how Bayesian deep learning can improve the performance of the machine–physician team in the skin lesion classification task.
We propose a DNN-based CAD model that uses approximate Bayesian inference to output
an uncertainty estimate along with its prediction in skin lesion classification.
They formulated metrics to evaluate the uncertainty estimation performance of the Bayesian models.
Accuracy
VGG16: 79.63%
ResNet50: 80.45%
DenseNet169: 81.35%
Bayesian VGG16: 81.02%
Bayesian ResNet50: 82.37%
Bayesian DenseNet169: 83.59%

1.4 GOAL

- To study the pre-trained models and classifiers using publically available datasets.
- To propose a model to improve the overall performance of multi-class skin classification and do a comparative study with pre-trained models.

CHAPTER II

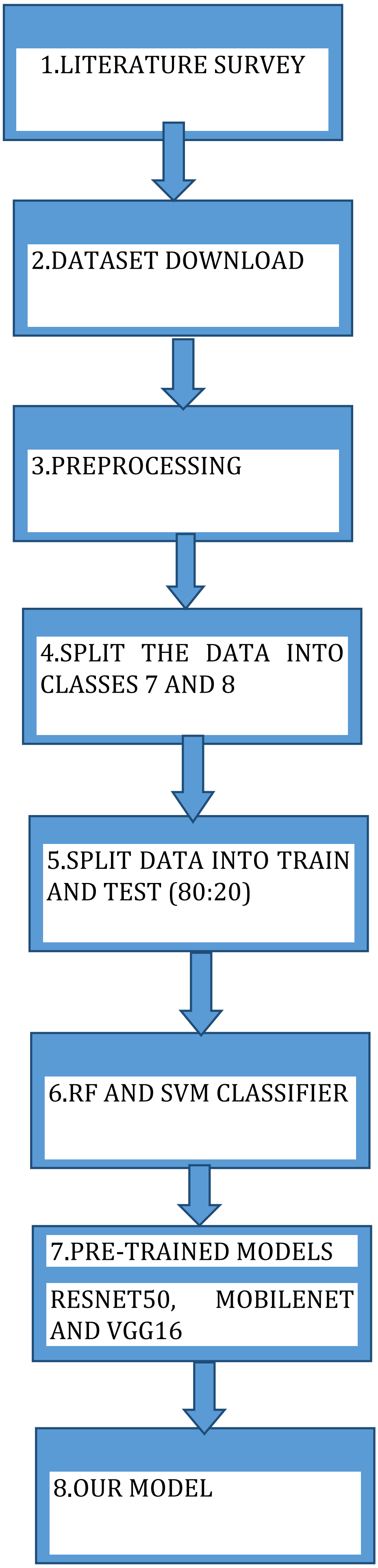
1. METHODOLOGY

Methodologies are a set of guiding principles and processes used to plan, manage, and execute projects. The project management methodology you choose determines how work is prioritized and completed.

In this section, we describe all the material and methods we've used for building the proposed system thoroughly including pre-trained models, data preparation, and model training.

The process we are using is Machine learning which is unquestionably increasing nowadays. The power of computers to find from examples rather than operating strictly in step with previously written rules is an exciting way of solving problems. [55] We are using the jupyter notebook (offline) platform for Machine Learning. We are using Python language to write codes, which is the most popular and preferred language for machine learning and data science. For more details, refer to chapter 4.

2.1FLOWCHART



2.1.1 LITERATURE SURVEY

The purpose of performing this literature review was to choose and categorize the foremost effective available approaches to carcinoma detection using Convolutional neural networks (CNNs).
Systematic literature reviews collect and analyze existing studies per predefined evaluation criteria. Such reviews help to determine what's already known within the concerned domain of study. All data collected from primary sources are organized and analyzed. Once systematic literature is completed, it provides a more sensible, logical, and robust answer to the underlying question of the research.
The population of studies considered within the present systematic literature review consisted of research papers relevant to SC detection supported deep neural network (DNN) techniques. [39]. Some details of the papers are given in chapter 1.

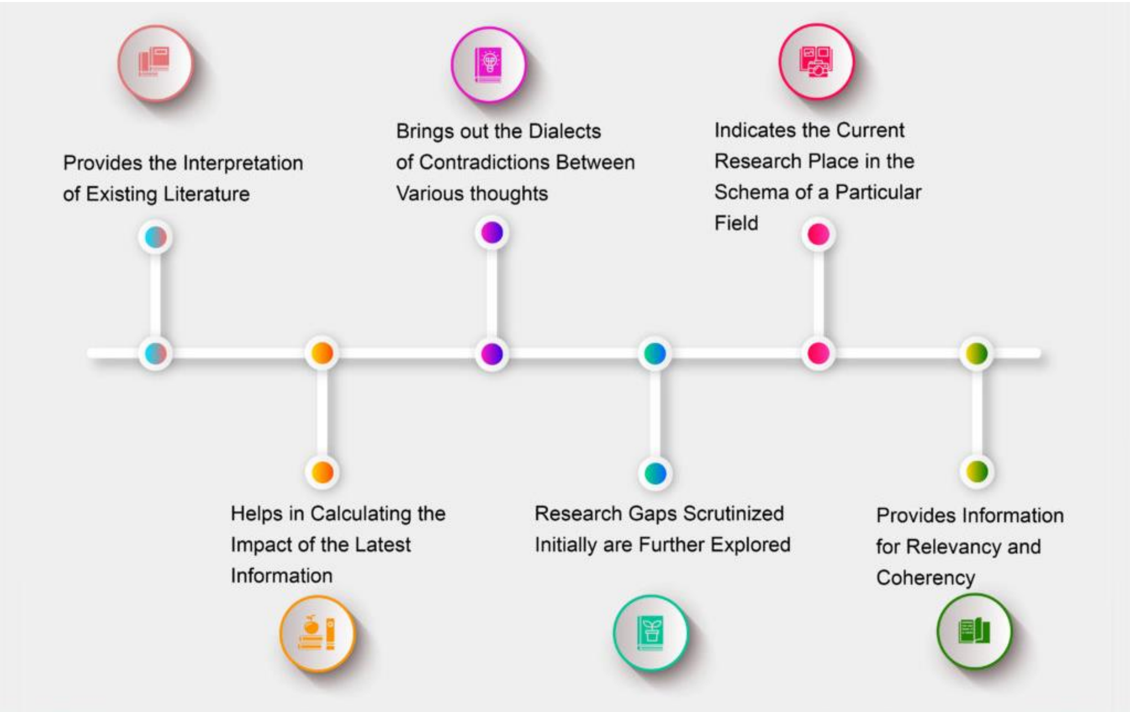


Figure 4

2.1.2 DATASET DOWNLOAD

Much scientific research relies on the gathering and analysis of measurement data. Scientific data sets are, at least, intermediate results for many research projects. For some time, data sets weren't even published and whether or not they were published it had been mostly as a not re-usable by-product of the publication. But a stimulating phenomenon may be observed here: data-sets (often along with models and parameters) are becoming more important themselves and will sometimes be seen because of the first intellectual output of the research. Publishing and preserving data sets should therefore seriously be considered. This could especially be the case if the data can't be reproduced (as they result from unique events) and may be necessary for future longitudinal research or to test or check future insights.

Good data provides indisputable evidence, while anecdotal evidence, assumptions, or abstract observation might cause wed resources because taking action supported an incorrect conclusion. Kaggle allows users to seek out and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to resolve data science challenges.

We downloaded two datasets: HAM10000 and ISIC 2019 which are publically available on the Kaggle platform. Details of these datasets are given in chapter 3.

2.1.3 PREPROCESSING

Data preprocessing in Machine Learning refers to the technique of preparing (cleaning and organizing) the information to form it suitable for building and training Machine Learning models.

Data Preprocessing is a Data processing technique that involves transforming data into an understandable format. Real-world data is usually incomplete, inconsistent, and/or lacking in certain behaviors or trends and is likely to contain many errors.

It is the primary and crucial step while creating a machine learning model. When creating a machine learning project, it's not always a case that we stumble upon clean and formatted data.

Preprocessing in Data Mining: Data preprocessing is a data mining technique that transforms the raw data into a useful and efficient format.

Steps Involved in Data Preprocessing:

- Data Cleaning
- Data Transformation
- Data Reduction

To lessen the computational cost of our proposed architecture, we resize and rescale the image size from 600 × 450 to 224 x 224 of the HAM10000 dataset. ISIC 2019 dataset is a combination of two datasets: HAM10000 and BCN_20000. BCN_20000 dataset contains images of size 1024 x 1024 which we resize to 224 x 224.

Details are given in chapter 3.

2.1.1 SPLIT THE DATA INTO CLASSES 7 AND 8

We have a very large folder of images, as well as a CSV file containing the class labels for each of those images. Because it is all in one giant folder, we have split them up into different folders for each class. For internal evaluation, we split the dataset into eight folders and seven folders of ISIC 2019 and HAM10000 respectively.

2.1.2 SPLIT DATA INTO TRAIN AND TEST (80:20)

We randomly split the data set into training (80%) and test set (20%) and used the same splits for training and testing the model. Details are given in chapter 3.

2.1.3 RF AND SVM CLASSIFIER

The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations based on training data. In Classification, a program learns from the given dataset or observations and then classifies new observations into several classes or groups. We selected SVM and RF for understanding purposes. Details are given in chapter 4.

2.1.4 PRE-TRAINED MODELS

There are several substantial benefits to leveraging pre-trained models:

- Super simple to incorporate.
- Achieve solid (same or even better) model performance quickly.
- There's not as much labeled data required.
- Versatile uses cases from transfer learning, prediction, and feature extraction.

Using a pre-trained model is significantly more accurate than using a custom-built convolutional neural network (CNN). Thus, it would make sense to start with a pre-trained model when doing image recognition tasks, as it is almost always the best course of action. We selected the 3 states of art models: Resnet50, MobileNet, and VGG16. Details are provided in chapter 4.

2.1.5 OUR MODEL

The main advantage of CNN compared to its predecessors is that it automatically detects the important features without any human supervision. A pre-trained model is a model created by someone else to solve a similar problem. Instead of building a model from scratch to solve a similar problem, you use the model trained on another problem as a starting point. Using the knowledge of pre-trained models, we created our won model. Details are given in chapter 4.

CHAPTER III

3.1 DATASET

Publicly available skin image datasets are increasingly used to develop machine learning algorithms for carcinoma diagnosis. However, the whole datasets and their respective content are currently unclear. This systematic review aimed to spot and evaluate all publicly available skin image datasets used for carcinoma diagnosis by exploring their characteristics, data access requirements, and associated image metadata. A combined MEDLINE, Google, and Google Dataset search identified 21 open-access datasets containing 106 950 skin lesion images, 17 open access atlases, eight regulated access datasets, and three regulated access atlases. Images and accompanying data from open access datasets were evaluated by two independent reviewers. Among the 14 datasets that reported country of origin, most (11 [79%]) originated from Europe, North America, and Oceania exclusively. Most datasets (19 [91%]) contained dermoscopic images or macroscopic photographs only. Clinical information was available regarding age for 81 662 images (76·4%), sex for 82 848 (77·5%), and body site for 79 561 (74·4%). Subject ethnicity data were available for 1415 images (1·3%), and Fitzpatrick skin type data for 2236 (2·1%). There was limited and variable reporting of characteristics and metadata among datasets, with substantial under-representation of darker skin types. This is the primary systematic review to characterize publicly available skin image datasets, highlighting limited applicability to real-life clinical settings and restricted population representation, precluding generalizability. Quality standards for characteristics and metadata reporting for skin image datasets are needed.

Digital health innovation has the potential to enhance health care by increasing access to specialist expertise. Among the myriad machine learning applications in health care, medical image classification, particularly for dermatology, has advanced substantially in recent years and includes the diagnosis of skin cancers from dermoscopic or macroscopic photographs. Advances in machine learning algorithm diagnostic accuracy have largely been driven by the use of deep learning architectures made possible through greater availability of computing power and huge repositories of digital images for algorithm training. For this purpose, large numbers of digital images easily accessible through publicly available datasets are utilized in dermatology. Publicly available datasets used for developing machine learning algorithms circumvent barriers to dataset procurement, like having appropriate technological infrastructure, regulatory approvals, time, and financial investment for large-scale digital image acquisition from participants. Furthermore, publicly available datasets are used as a benchmark for direct comparison of algorithm performance.

Skin cancer incidence continues to rise globally, placing increasing demands on healthcare services. Digital solutions to address this demand are reflected by accelerated tele dermatology adoption during the COVID-19 pandemic. Machine learning algorithms have the potential for automated diagnosis of skin malignancies through digital image analysis, and the diagnostic accuracy of machine learning algorithms has been shown within the past 5 years to be comparable to, or even surpass, dermatologists in controlled experimental settings.

Publicly available skin image datasets, like those hosted through the International Skin Imaging Collaboration (ISIC) archive, are increasingly used to develop machine learning algorithms for carcinoma diagnosis. Additionally, although primarily aimed to be used as educational resources, dermatology atlases (compilations of photographs of skin diseases) containing digital images are frequently used as a source of skin lesion images for algorithm development. However, with training data from circumscribed populations, often curated retrospectively, machine learning algorithms are at risk of over-fitting, and their generalizability is heavily influenced by the participants and images used for training, which are vulnerable to selection bias. Algorithms used for skin lesion classification frequently underperform when tested on independent datasets. Further examples of the susceptibility of machine learning algorithms to biases for clinical factors such as age, sex, ethnicity, and socioeconomic status have also been reported in diverse areas of health care and computer science. This underlines the importance of detailing the precise composition of datasets through metadata reporting, to confirm the generalizability of algorithms to real-world populations. Furthermore, the recently developed concept of health data poverty—a systematic data disparity resulting in inequalities in health care emphasizes the necessity to make sure diversity, transparency, and usefulness of datasets.

Demographic and clinical metadata also can greatly influence machine learning algorithm development and validation, with classification accuracy increasing for skin lesions when subject and lesion metadata are integrated into models. This more closely reflects clinical practice, as dermatologists conduct in-person history taking and evaluation of skin lesions via the naked eye, dermoscopic, and physical examination (e.g., lesion palpation), and not through image analysis alone. Completeness of metadata provides information about the population, disease, and data types on which the algorithm was trained or validated, which is important for extrapolating assumptions of generalizability of algorithm performance to other populations.

To the most effective of our knowledge, there aren't any guidelines or quality standards that characterize optimal skin image datasets that would be used to train machine learning algorithms and these datasets haven't previously been systematically characterized. This review aimed to spot publicly available skin image datasets used to develop machine learning algorithms for carcinoma diagnosis, categorize their data access requirements, and systematically evaluate their characteristics including associated metadata.

Machine learning for cancer detection isn't possible without data. Unfortunately, there are only a few ready-to-use datasets for training a neural network to classify skin lesions. However, we managed to find the HAM10000 dataset and ISIC 2019 dataset which are decent dataset that contains a large collection of multi-source dermatoscopic images of common pigmented skin lesions. We downloaded both datasets from the Kaggle platform.

3.1.1. HAM10000

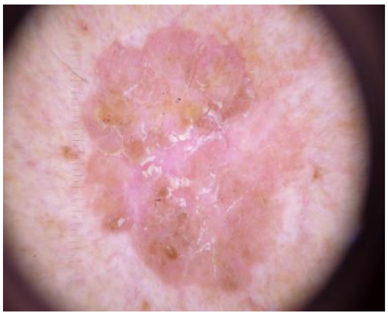
The HAM10000 training set contains pigmented lesions from different populations. The Austrian image set consists of lesions of patients referred to a tertiary European referral center specialized in early detection of melanoma in high-risk groups. This group of patients often has a high number of nevi and a personal or family history of melanoma. The Australian image set includes lesions from patients of a primary care facility in a high skin cancer incidence area. Australian patients are typified by severe chronic sun damage. Chronic sun-damaged skin is characterized by multiple solar lentigines and ectatic vessels, which are often present within the periphery of the target lesion. Very rarely also small angiomas and seborrhoea Kerasotes may impinge on the target lesion.

Dermatoscopic images of both study sites were taken by different devices using polarized and non-polarized dermatoscopy. The set includes representative samples of pigmented skin lesions that are practically relevant. Over 95% of all lesions encountered during clinical practice will make up one of the seven diagnostic categories. In practice, the task of the clinician is to differentiate between malignant and benign lesions, but also to make specific diagnoses because different malignant lesions, for example, melanoma and basal cell carcinoma, may be treated in a different way and timeframe. Except for vascular lesions, which are pigmented by hemoglobin and not by melanin, all lesions have variants that are completely devoid of pigment (for example amelanotic melanoma). Non-pigmented lesions, which are more diverse and have a larger number of possible differential diagnoses, are not in this set.

HAM10000 dataset is a benchmark dataset with over 50% of lesions confirmed by pathology. The dataset consists of a total of 10015 dermoscopy images, which includes 6705 Melanocytic nevi images(nv), 1113 Melanoma images(mel), 1099 Benign keratosis images(bkl), 514 Basal cell carcinoma images(bcc), 327 Actinic keratosis images(akiec), 142 Vascular images(vasc) and 115 Dermatofibroma images(df) with 600X450 pixels resolution.

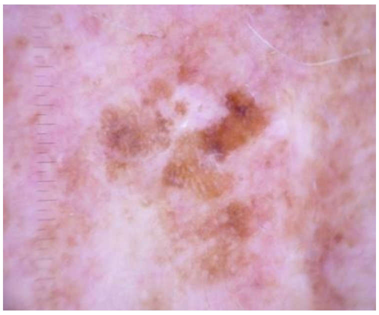
3.1.2. Sample images of skin cancer types from HAM10000 are represented below.

(a) Actinic Keratosis(akiec)



(b) Basal Cell Carcinoma(bcc)

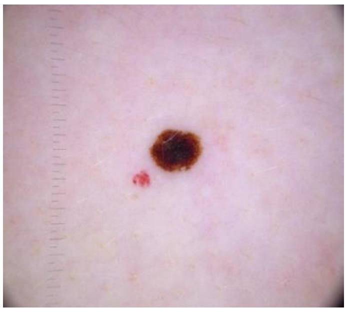
(c) Benign Keratosis(bkl)



(d) Dermatofibroma(df)

(e) Melanocytic nevi (nv)

(f) Melanoma(me)



(g) Vascular Lesions

3.1.3. Description of diagnostic categories

1. Akiec

Actinic Keratoses (Solar Keratoses) and Intraepithelial Carcinoma (Bowen’s disease) are common noninvasive, variants of squamous cell carcinoma that can be treated locally without surgery. Some authors regard them as similar to squamous cell carcinomas and not as actual carcinomas. There is, however, an agreement that these lesions may progress to invasive squamous cell carcinoma – which is usually not pigmented. Both neoplasms commonly show surface scaling and commonly are devoid of pigment. Actinic keratoses are more common on the face and Bowen’s disease is more common on other body sites. Because both types are induced by UV light the surrounding skin is usually typified by severe sun damage except in cases of Bowen’s disease that are caused by human papillomavirus infection and not by UV. Pigmented variants exist for Bowen’s disease20 and for actinic keras21, and both are included in this set. The dermatoscopic criteria of pigmented actinic keratoses and Bowen’s disease are described in detail by Zalaudek et al. [22,23] and by Cameron et al. [20]

2. Bcc

Basal cell carcinoma is a common variant of epithelial carcinoma that rarely metastasizes but grows destructively if untreated. It appears in numerous morphologic variants (flat, nodular, pigmented, cystic), which are described in additional detail by Lallas et al. [24]

3. Bkl

"Benign keratosis" is a generic class that has seborrheic keratoses ("senile wart"), solar lentigo - which might be considered as a flat variant of seborrheic keratosis - and lichen-planus like keratoses (LPLK), which corresponds to a seborrheic keratosis or a solar lentigo with inflammation and regression [25]. The three subgroups may look different dermatoscopically, but we grouped them together because thilar biologically and sometimes reported under the identical generic term histopathologically. From a dermatoscopic view, lichen planus-like keratosis is incredibly challenging because it can show morphologic features mimicking melanoma [26] and are oftenisopsied or excised for diagnostic reasons. The dermatoscopic appearance of seborrheic keratosis varies in line with anatomic site and sort. [27]

4. Df

Dermato fibroma is a benign skin lesion considered as either a benign proliferation or an inflammatory reaction to minimal trauma. The foremost common dermatoscopic presentation is reticular lines at the periphery with a central white patch denoting fibrosis. [28].

5. Nv

Melanocytic nevi are benign neoplasms of melanocytes and appear during a myriad of variants, which all are included in our series. The variants may differ significantly from a dermatoscopic point of view. In contrast to melanoma, they're usually symmetric iaboutthe distribution of color and structure. [29].

6. Mel

Melanoma is a malignant neoplasm derived from melanocytes which will appear in numerous variants. If excised in an early stage it can be cured by simple surgical excision. Melanomas can be invasive or noninvasive (in situ). We included all variants of melanoma including melanoma in situ but did exclude non-pigmented, subungual, ocular, or mucosal melanoma. Melanomas are usually, albeit not always, chaotic, and a few melanoma-specific criteria rely upon anatomic sites [23,30].

7. vasc

Vascular skin lesions within the dataset range from cherry angiomas to angiokeratomas31 and pyogenic granulomas [32]. Hemorrhage is additionally included in this category. Angiomas are dermatoscopically characterized by red or purple color and solid, well-circumscribed structures called red clods or lacunes. The number of images within the datasets doesn't correspond to the number of unique lesions, because we also provide images of the identical lesion taken at different magnifications or angles, or with different cameras. This should function as a natural data augmentation because it shows random transformations and visualizes both general and native features.

Table 2

Dataset	**Total Images**	**AKIEC**	**BCC**	**BKL**	**DF**	**MEL**	**NV**	**VASC**
HAM10000	**10015**	**327**	**514**	**1099**	**115**	**1113**	**6705**	**142**

3.1.4. ISIC2019

ISIC 2019 (Tschandl, Rosendahl and Kittler, 2018; Codella, Gutman, Celebi, Helba, Marchetti, Dusza, Kalloo, Liopyris, Mishra, Kittler et al., 2018; Combalia, Codella, Rotemberg, Helba, Vilaplana, Reiter, Carrera, Barreiro, Halpern, Puig et al., 2019), the large-scale publicly available dermoscopy imaging database, that contains 25331 dermoscopic images acquired from different resources together with their diagnosis labels as their ground truth. A part of the images also has corresponding Meta information, like gender, age, anatomy site, etc. The ISIC 2019 contains 8 different skin diseases including malignant melanoma (MM), melanocytic nevus (MN), basal cell carcinoma (BCC), actinic keratosis (AK), benign keratosis (BKL), dermatofibroma (DF), vascular lesion (VASC) and squamous cell carcinoma (SCC). Part of the training dataset is the HAM10000 dataset which contains images of size 600 × 450 that was centered and cropped around the lesion. The dataset curators applied histogram corrections to some images [1]. Another dataset, BCN_20000, contains images of size 1024 × 1024. This dataset is especially challenging as many images are uncropped and lesions in difficult and uncommon locations are present [33]. Last, the MSK dataset contains images of assorted sizes.

The dataset also contains meta-information about the patient’s age group (in steps of 5 years), the anatomical site (eight possible sites), and also the sex (male/female). The metadata is partially incomplete, i.e., there are missing values for a few images.

For internal evaluation, we split the main training dataset into eight folders. The dataset contains multiple images of identical lesions. Thus, we make sure that all images of the identical lesion are within the same folder. Note that we don’t include any of our images from the unknown class in our evaluation as we don’t know whether or not they accurately represent the particular unknown class. Thus, all our models are trained to predict nine classes but we only evaluate the known, eight classes.

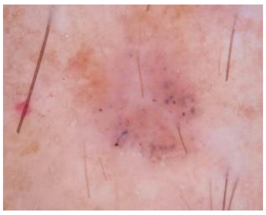
Table3

Dataset	NV	M	BKL	BCC	SCC	VL	DF	AK	Total
ISIC 2019	12,875	4522	2624	3323	628	253	239	867	25,331

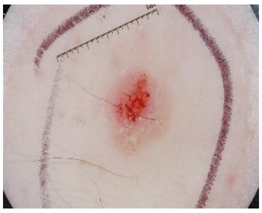
3.1.5. Sample images of skin cancer types from ISIC 2019 are represented below.

(a) Actinic Keratosis (akiec)

(b) Basal Cell Carcinoma(bcc)



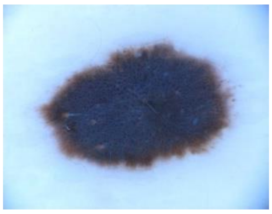
(c) Benign Keratosis(bkl)



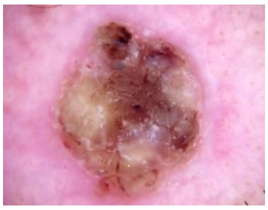
(d) Dermatofibroma (df)

(f) Melanoma(me)

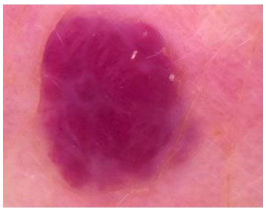
(e)Melanocytic nevi (nv)



(f)Squamous cell carcinoma (scc)



(g) Vascular Lesions(vasc)



3.2PREPROCESSING

The first and most important step in developing an automatic detection system is the preprocessing step. [1] The preprocessing phase aims at ameliorating the dermoscopic image's quality to confirm improved lesion detection efficiency (Fig. 4). Throughout image acquisition, the presence of artifacts like hair, bad lighting, and other noise may result in an imperfect diagnosis. There are common artifact removal algorithms like DullRazor (Lee et al., 1997), which are employed to extract hair artifacts. The algorithm uses morphological filters to detect the position of the hair and replace the detected hair with its neighboring pixels. Adaptive median filters are used to smooth out the final image. We can also apply various filters for removing noises (Hoshyar and Al-Jumaily, 2014). In addition, to further improve the image quality, some image enhancement methods are also used. The foremost important of them is color correction or calibration (Wighton et al., 2011). We can find other techniques like illumination correction, contrast enhancement, and edge enhancement (Maglogiannis et al., 2006). Readers will find a summary of preprocessing methods related to dermoscopy images in Celebi et al. (2015).

3.2.1. SPLIT INTO FOLDERS

The HAM10000 dataset is split into 7 folders of classes as shown below**:**

Table 4

CLASS	**BCC**	**BKL**	**DF**	**MEL**	**NV**	**VASC**	**AKIEC**
NO. OF IMAGES	**514**	**1099**	**115**	**1113**	**6705**	**142**	**327**

The ISIC 2019 dataset is split into 8 folders of classes as shown below:

Table 5

CLASS	**BCC**	**BKL**	**DF**	**MEL**	**NV**	**VASC**	**AKIEC**	**SCC**
NO. OF IMAGES	**3323**	**2624**	**239**	**4522**	**12875**	**253**	**867**	**628**

3.2.2. TRAIN-TEST SPLIT

The train-test split procedure is employed to estimate the performance of machine learning algorithms after they are used to make predictions on data not used to train the model. It is a quick and simple procedure to perform, the results of which permit you to check the performance of machine learning algorithms for your predictive modeling problem. The train-test split procedure is used after you have a very large dataset, a costly model to train, or require a good estimate of model performance quickly.

The procedure involves taking a dataset and dividing it into two subsets. The primary subset is employed to fit the model and is referred to as the training dataset. The second subset is not used to train the model; instead, the input element of the dataset is provided to the model, then predictions are made and compared to the expected values. This second dataset is referred to as the test dataset.

Train Dataset: Used to fit the machine learning model.
Test Dataset: Used to evaluate the fit machine learning model.

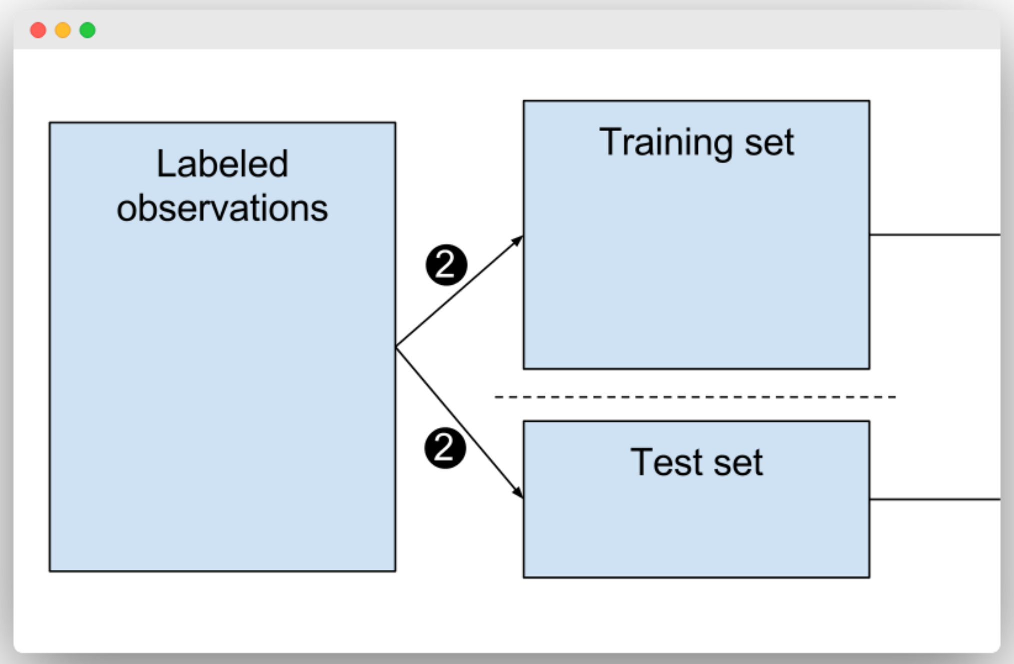


Figure 5

The idea of “sufficiently large” is specific to each predictive modeling problem. It means that there is enough data to split the dataset into train and test datasets and each of the train and test datasets are a suitable representation of the problem domain. This requires that the original dataset is also a suitable representation of the problem domain.
Conversely, the train-test procedure is not appropriate when the dataset available is small. The reason is that when the dataset is split into train and test sets, there will not be enough data in the training dataset for the model to learn an effective mapping of inputs to outputs. There will also not be enough data in the test set to effectively evaluate the model performance. The estimated performance could be overly optimistic (good) or overly pessimistic (bad). In addition to dataset size, another reason to use the train-test split evaluation procedure is computational efficiency. Alternately, a project may have an efficient model and a vast dataset, although may require an estimate of model performance quickly. Again, the train-test split procedure is approached in this situation. Samples from the original training dataset are split into two subsets using random selection. This is to ensure that the train and test datasets are representative of the original dataset. The procedure has one main configuration parameter, which is the size of the train and test sets. This is most commonly expressed as a percentage between 0 and 1 for either the train or test datasets. For example, a training set with the size of 0.67 (67 percent) means that the remaining percentage of 0.33 (33 percent) is assigned to the test set. There is no optimal split percentage. You must choose a split percentage that meets your project’s objectives with considerations that include:

- Computational cost in training the model.
- Computational cost in evaluating the model.
- Training set representativeness.
- Test set representativeness.

Nevertheless, common split percentages include:

- Train: 80%, Test: 20%
- Train: 67%, Test: 33%
- Train: 50%, Test: 50%

We have split the data into 80% train set and 20% validation set. Scientists want to do predictions create a model and test the data. When they do that, two things can happen over-fitting and under-fitting.

3.2.2. OVERFITTING

Overfitting is most common than Under-fitting, but none should happen in order to avoid affecting the predictability of the model. Overfitting can happen when the model is too complex. Overfitting means that the model we trained has trained "too well" and fits too closely to the training dataset. The problem is that the accuracy of the training data will unable accurate for untrained or new data. To avoid it, the data can't have many features/variables compared to the number of observations.

3.2.4. UNDERFITTING

Under-fitting can happen when the model is too simple and means that the model does not fit the training data. To avoid it, the data need enough predictors/independent variables.

3.2.5. VALIDATION

Cross-Validation is when scientists split the data into (k) subsets, and train on k-1 one of those subsets. The last subset is the one used for the test. Some libraries are most commonly used to do training and testing.

- Pandas: used to load the data file as a Pandas data frame and analyze it.
- Sklearn: used to import the datasets module, load a sample dataset and run a linear regression.
- Matplotlib: using pyplot to plot graphs of the data.

If you need to split the database, first avoid Overfitting or Under-fitting. When we split the dataset into training and testing, it is as follows:

- One has independent features, called (x).
- One has dependent variables, called (y).

To split it, we do:

x Train – x Test / y Train – y Test
x Train and y Train become data for machine learning, capable to create a model. Once the model is created, input x Test and the output should be equal to y Test. The more closely the model output is to the y Test: the more accurate the model is. X corresponds to your float feature matrix of shape (n_samples, n_features) (aka. the design matrix of your training set). y is the float target vector of shape (n_samples) (the label vector). In our case, label 0 could correspond to a spam example, and 1 to a ham one.

1). X_train - This includes your all-independent variables; these will be used to train the model. About 60% of observations from your complete data will be used to train/fit the model and the rest 40% will be used to test the model.
2). X_test - This is the remaining 40% portion of the independent variables from the data which will not be used in the training phase and will be used to make predictions to test the accuracy of the model.
3). y_train - This is your dependent variable that needs to be predicted by this model, this includes category labels against your independent variables, we need to specify our dependent variable while training/fitting the model.
4). y_test - This data has category labels for your test data, these labels will be used to test the accuracy between actual and predicted categories. [56]

Total number of images in ISIC 2019 dataset: 25,331

Total number of images after augmentation: 30,320

Of the total set of 30,320 dermatoscopic images, 25251images were used for training and 5069 images for testing.

Table of split data ISIC 2019

Diagnosis	Total	train	test
Mel	5417	4512	905
Bcc	3978	3313	665
Bkl	3139	2614	525
Ak	1031	857	174
Nv	15440	12,865	2575
Vasc	324	243	51
Df	277	229	48
Scc	744	618	126

Table 6

Total number of images in HAM10000 dataset: 10,015
Total number of images after augmentation: 11,950

Of the total set of 11,950 dermatoscopic images, 9945 images were used for training and 2005 images for testing.

Table of split data HAM1000

Diagnosis	Total	train	test
Mel	1326	1103	223
Bcc	607	504	103
Bkl	1309	1089	220
Akiec	383	317	66
Nv	8036	6695	1341
Vasc	161	132	29
df	128	105	23

Table 7

3.2.6. DATA AUGMENTATION

When you train a machine learning model, what you're doing is tuning its parameters such that it can map a specific input i.e. image to some output i.e. a label. Our optimization goal is to chase that sweet spot where our model's loss is low, which happens when your parameters are tuned in the right way.
Naturally, if you have plenty of parameters, you'd have to show your machine learning model a proportional amount of examples, to induce good performance. Also, the number of parameters you need is proportional to the complexity of the task your model has to perform.

You don't need to explore novel new images which can be added to your dataset. Because neural networks aren't smart to start with. For example, a poorly trained neural network would think that these three tennis balls shown below are distinct, unique images.

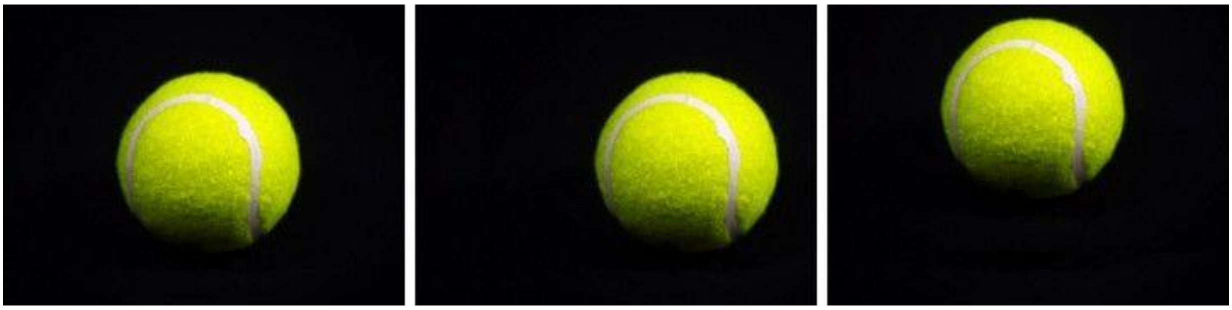


Figure 6

So, to get more data, we just need to make minor alterations to our existing dataset. Minor changes such as flips or translations or rotations. Our neural network would think these are distinct images anyway.

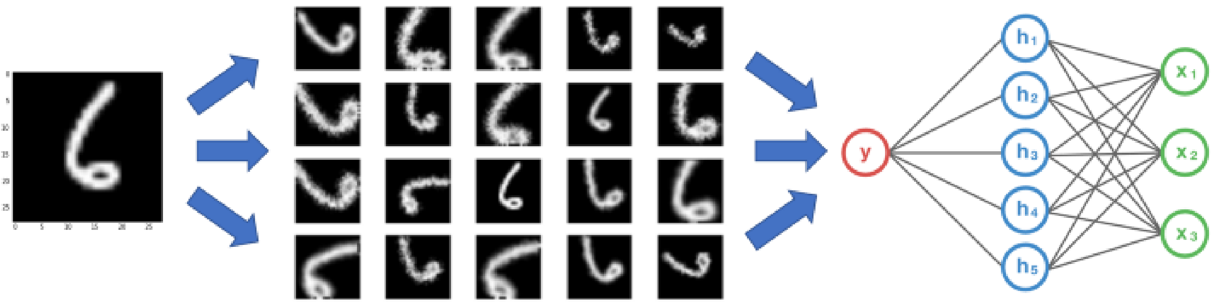


Figure 7

A convolutional neural network that can robustly classify objects even if it is placed in several orientations is said to have the property called invariance. More specifically, a CNN is often invariant to translation, viewpoint**,** size, or illumination. This essentially is the premise of data augmentation. In the real global scenario, we may have a dataset of images taken in a limited set of conditions. But, our target application may exist in a variety of conditions, like a different orientation, location, scale, brightness, etc. We account for these situations by training our neural network with additional synthetically modified data.

It can help to increase the amount of relevant data in the dataset. It is related to the way with which Neural networks learn. We have to reduce unwanted or irrelevant features in the dataset. You can just flip the images horizontally such that they face the other side.

There are two options you can do

1. To perform all necessary transformations beforehand, essentially increasing the size of your dataset.
2. To perform the transformations on a small batch, just before feeding it to your model.

Deep learning models require a decent amount of data to produce good results [48], so we use different data augmentation processes to train our model. We employ horizontal flipping with a probability of 0.50 to transform the images. Then, the zoom range (-0.2, +0.2) was used to randomly zoom in inside.
The random transformations used on images were:

Rescale = 1. /255,

Shear range = 0.2,

Zoom range = 0.2,

Horizontal flip = True,

CHAPTER IV

4. ALGORITHM

Machine learning (ML) is the study of computer algorithms that improve automatically through experience and by the use of data. It's seen as part of AI. Machine learning algorithms build a model based on sample data, called training data, to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are employed in a wide variety of applications, like in medicine, email filtering, speech recognition, and computer vision, where it's difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

Figure 8

A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers; but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory, and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain. In its application across business problems, machine learning is also referred to as predictive analytics.

Machine learning programs can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks. For simple tasks assigned to computers, it is possible to program algorithms telling the machine how to execute all steps required to solve the problem at hand; on the computer's part, no learning is needed. For more advanced tasks, it can be challenging for a human to manually create the needed algorithms. In practice, it can turn out to be more effective to help the machine develop its algorithm, rather than having human programmers specify every needed step.

The discipline of machine learning employs various approaches to teach computers to accomplish tasks where no fully satisfactory algorithm is available. In cases where vast numbers of potential answers exist, one approach is to label some of the correct answers as valid. This can then be used as training data for the computer to improve the algorithms it uses to determine correct answers. For example, to train a system for the task of digital character recognition, the MNIST dataset of handwritten digits has often been used.

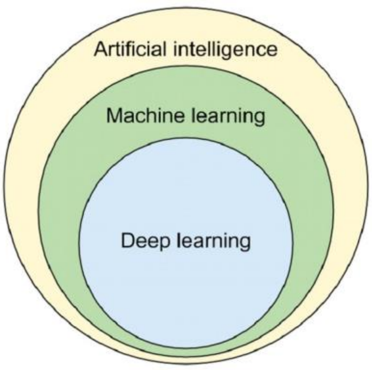


Figure 9

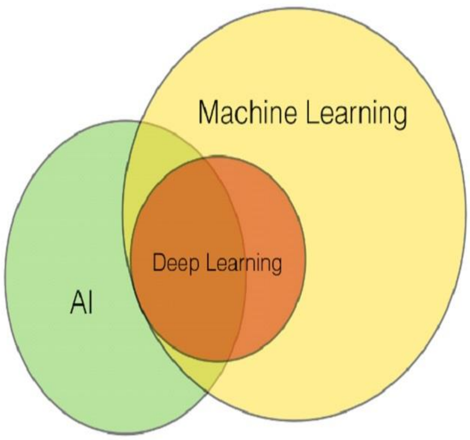


Figure 10

Machine learning as a subfield of AI   Part of machine learning as a subfield of AI

4.1 The process of machine learning can be split into the following steps:

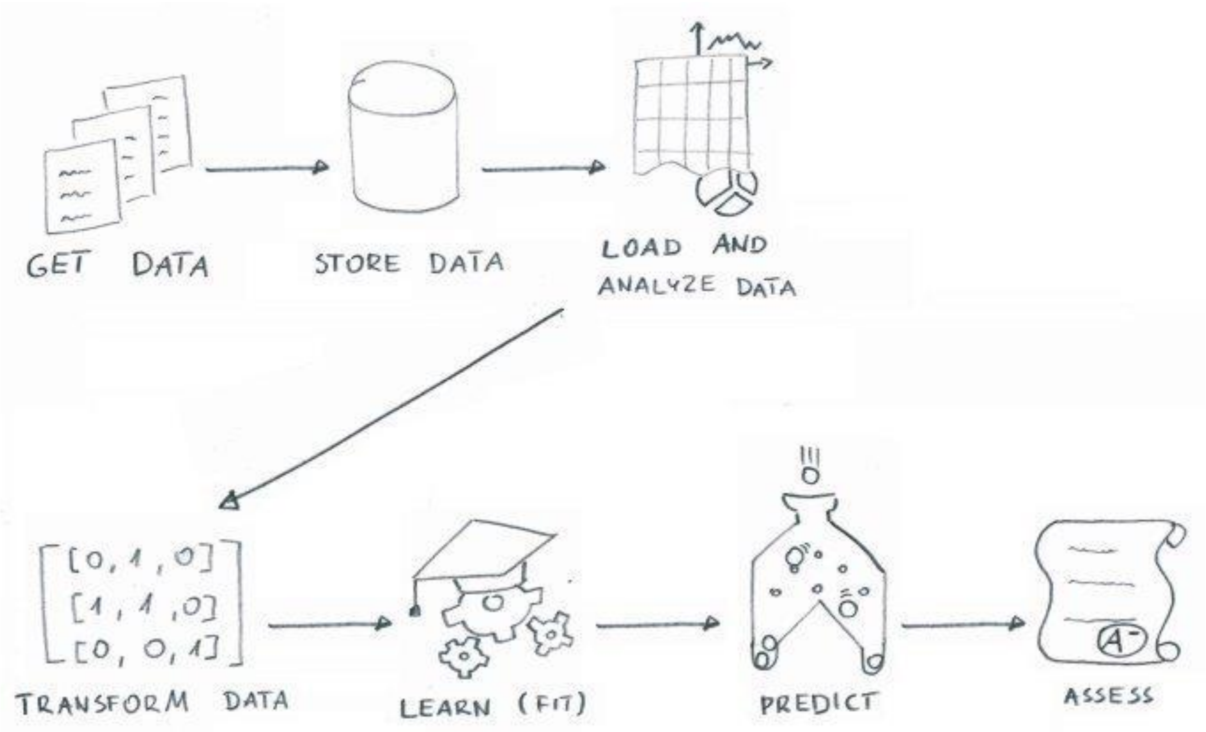


Figure 11

a) Get data

To start the machine learning process, you need to possess a set of data to be used for training the algorithm. Acquire a big enough dataset (including labels or answers to your problem). It's very important to ensure that the source of data is credible, otherwise, you would receive incorrect results, even if the algorithm itself is working correctly.

The second important thing is the size of the dataset. There is no straightforward answer for how large it should be. The answer may depend on many factors, for example:

- the type of problem you're looking to solve,
- the number of features in the data,
- the type of algorithm used. [55]

b) Store data

Store the acquired data in a single location for easy retrieval.

c) Load and analyze data

Load your dataset from storage and do basic data analysis and visualization.

d) Transform data

Machine learning requires purely numeric input, so you need to transform the input data.

e) Learn (fit)

Run the labeled data through a machine learning algorithm yielding a model.

f) Predict

Use the model to predict labels for data that the model did not see previously.

g) Assess

Verify the accuracy of predictions made by the model.

Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system:

- Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
- Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
- Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize supervised learning.

4.1.1. Supervised learning:

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs. The data is known as training data and consists of a set of training examples. Each training example has one or more inputs and the desired output, also known as a supervisory signal. In the mathematical model, each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix. Through iterative optimization of an objective function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs. An optimal function will allow the algorithm to correctly determine the output for inputs that were not a part of the training data. An algorithm that improves the accuracy of its outputs or predictions over time is said to have learned to perform that task. Types of supervised-learning algorithms include active learning, classification, and regression.

Classification algorithms are used when the outputs are restricted to a limited set of values, and regression algorithms are used when the outputs may have any numerical value within a range. As an example, for a classification algorithm that filters emails, the input would be an incoming email, and the output would be the name of the folder in which to file the email. Similarity learning is an area of supervised machine learning closely related to regression and classification, but the goal is to learn from examples using a similarity function that measures how similar or related two objects are. It has applications in ranking, recommendation systems, visual identity tracking, face verification, and speaker verification.[33]

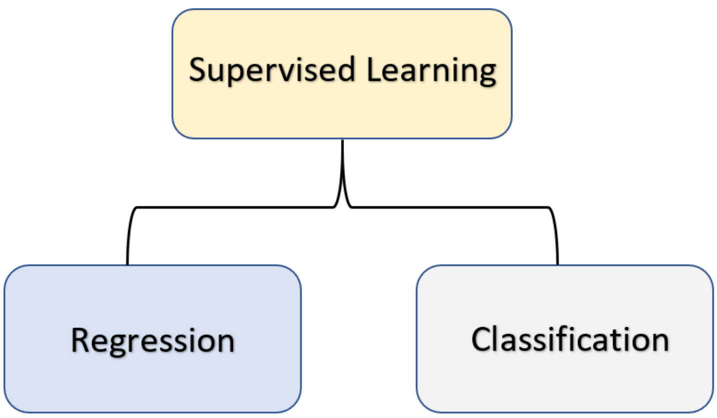


Figure 12

4.1.2. Unsupervised learning

Unsupervised learning algorithms take a set of data that contains only inputs, and fine structure the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labeled, classified, or categorized. Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data. A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function. Though unsupervised learning encompasses other domains involving summarizing and explaining data features.

Cluster analysis is the assignment of a set of observations into subsets (called clusters) so that observations within the same cluster are similar according to one or more predestinated criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some similarity metric and evaluated, for example, by internal compactness, or the similarity between members of the same cluster, and separation, the difference between clusters. Other methods are based on estimated density and graph connectivity.

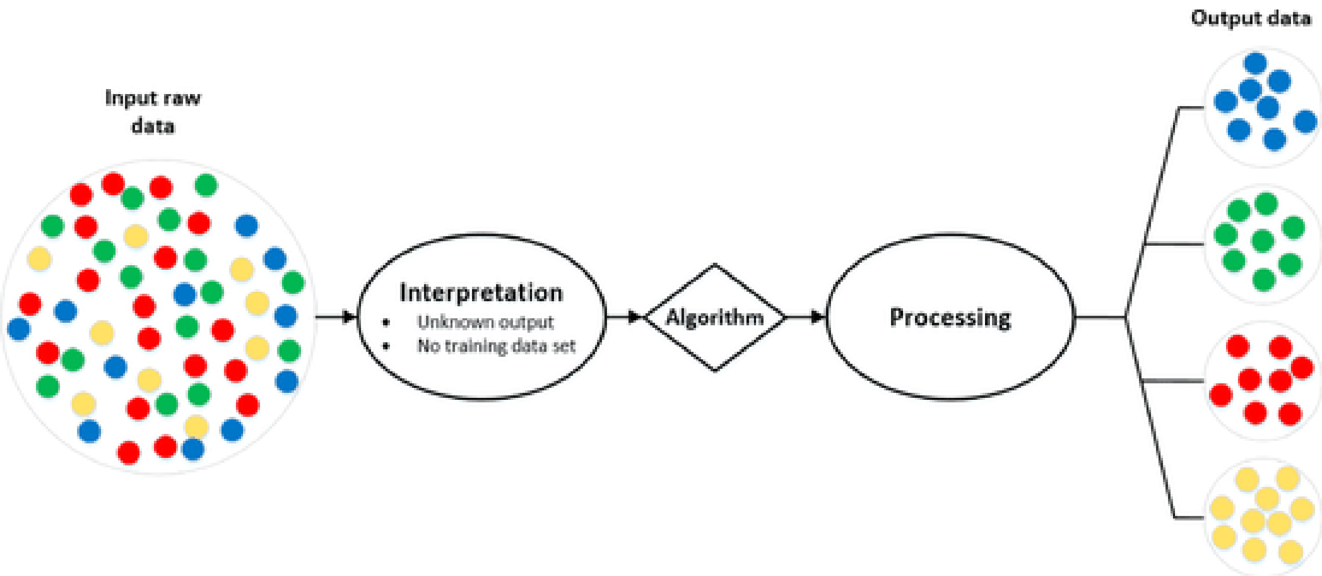


Figure 13

4.1.3. Semi-supervised learning:

Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Some of the training examples are missing training labels, yet many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce a considerable improvement in learning accuracy.

In weakly supervised learning, the training labels are noisy, limited, or imprecise; however, these labels are often cheaper to obtain, resulting in larger effective training sets. [33]

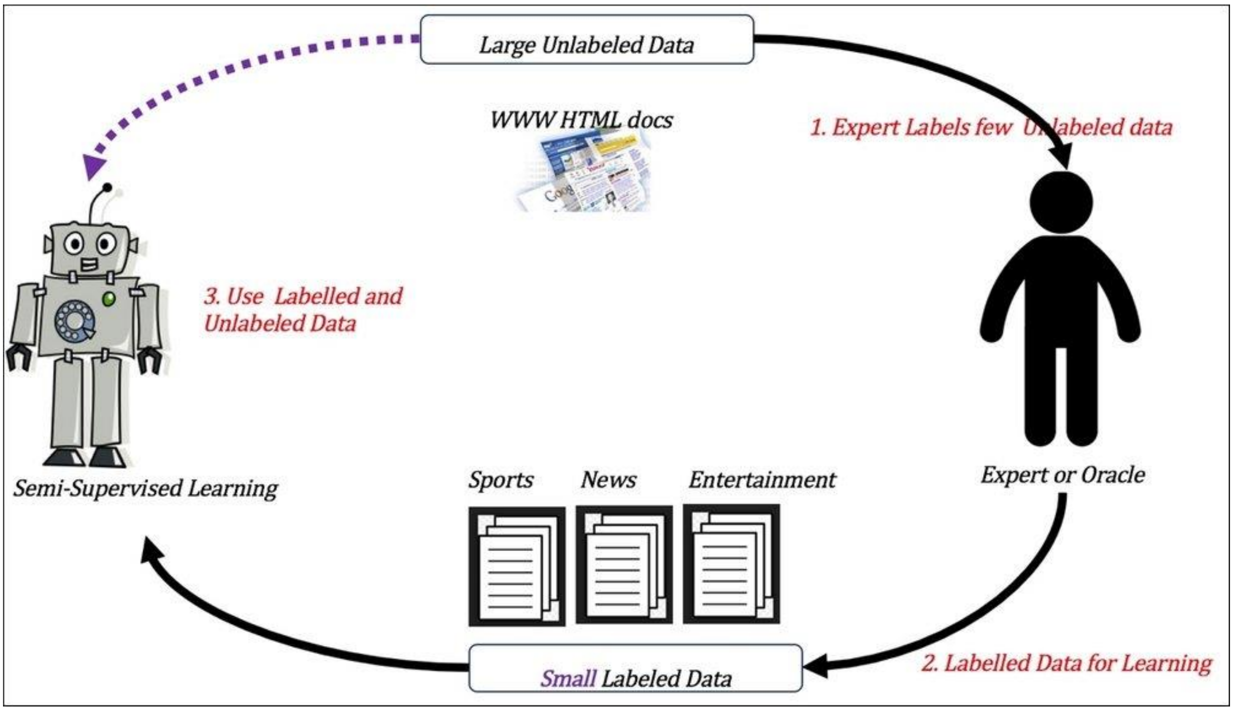


Figure 14

4.1.4. Reinforcement learning

Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment to maximize some notion of cumulative reward. Due to its generality, the field is studied in many other disciplines, such as game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent, systems, swarm intelligence, statistics and g,enetic algorithms. In machine learning, the environment is typically represented as a Markov decision process (MDP). Many reinforcement learning algorithms, use dynamic programming techniques. Reinforcement learning algorithms do not assume knowledge of an exact mathematical model of the MDP and are used when exact models are infeasible. Reinforcement learning algorithms are used in autonomous vehicles or in learning to play a game against a human opponent. [33]

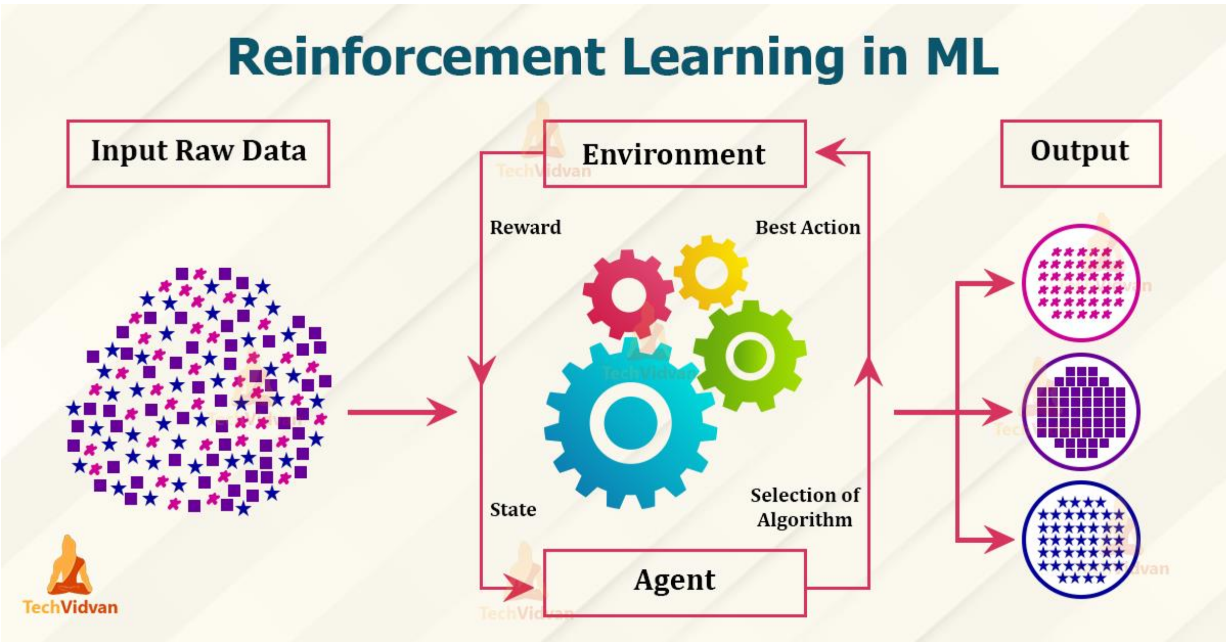


Figure 15

4.2. BLOCK DIAGRAM OF ML

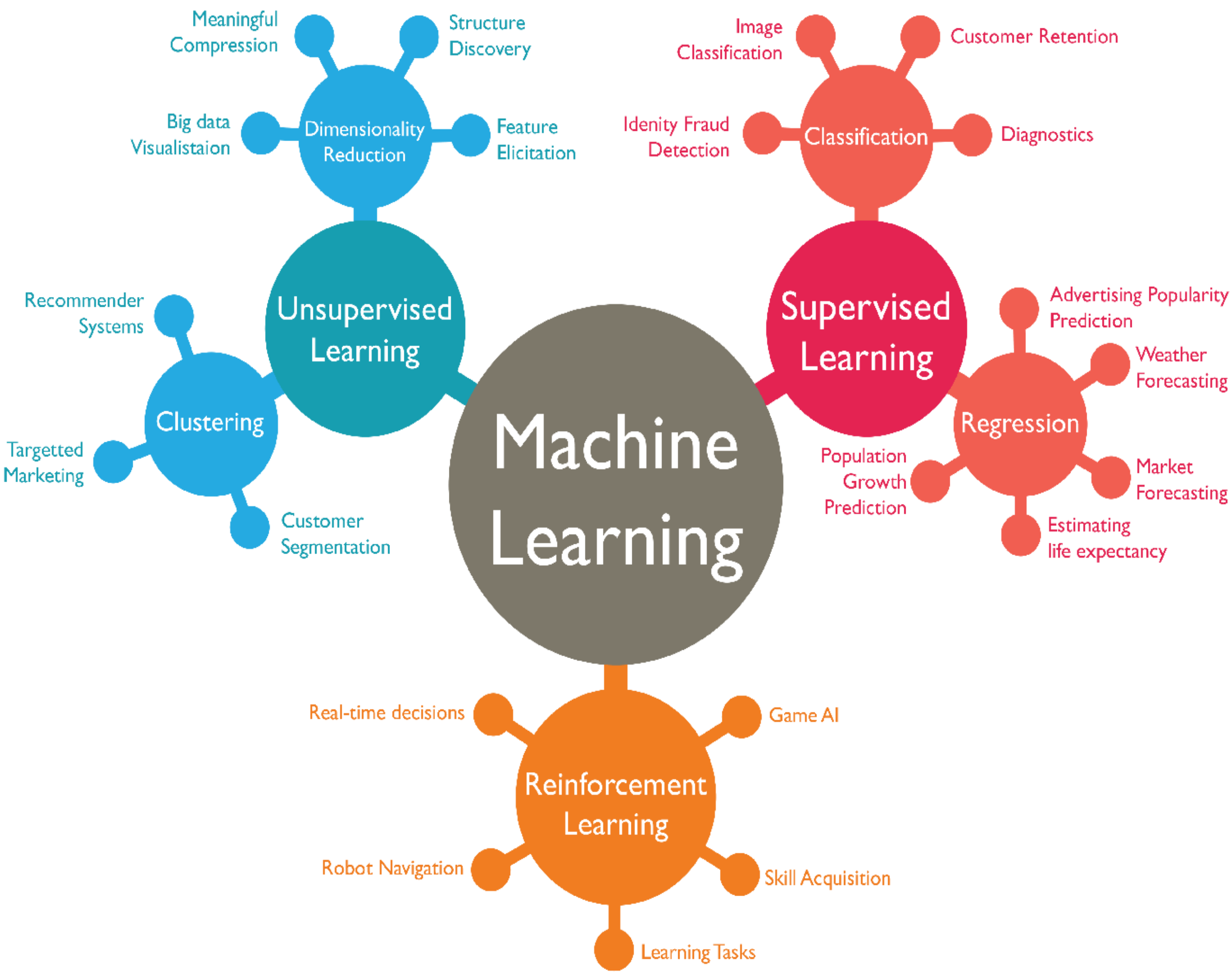


Figure 16

Our Algorithm comes under supervised learning because we are doing image classification.

4.2.1. IMAGE CLASSIFICATION

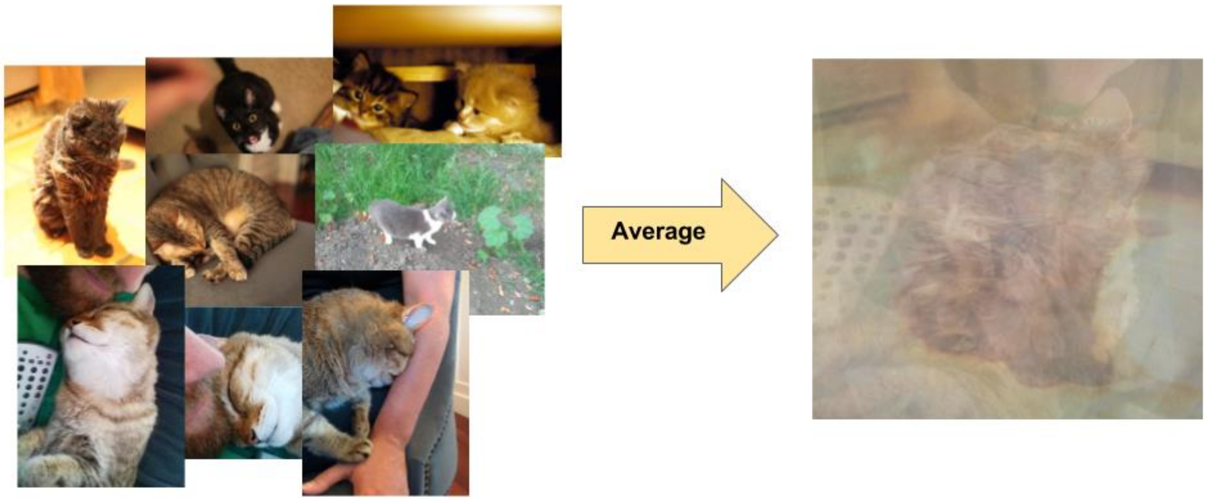


Figure 17

To model objects more flexibly, classic computer vision models added new features derived from pixel data, such as color histograms, textures, and shapes. The downside of this approach was that feature engineering became a real burden, as there were so many inputs to tweak. For a cat classifier, which colors were most relevant? How flexible should the shape definitions be? Because features needed to be tuned so precisely, building robust models was quite challenging, and accuracy suffered. [57]

We live in the era of data. With the Internet of Things (IoT) and Artificial Intelligence (AI) becoming ubiquitous technologies, we now have huge volumes of data being generated. Differing in form, data could be speech, text, image, or a mix of any of these. In the form of photos or videos, images make up for a significant share of global data creation. Since the vast amount of image data we obtain from cameras and sensors is unstructured, we depend on advanced techniques such as machine learning algorithms to analyze the images efficiently. Image classification is probably the most important part of digital image analysis. It uses AI-based deep learning models to analyze images with results that for specific tasks already surpass human-level accuracy (for example, in face recognition). Since AI is computationally very intensive and involves the transmission of huge amounts of potentially sensitive visual information, processing image data in the cloud comes with severe limitations. Therefore, there is a big emerging trend called Edge AI that aims to move machine learning (ML) tasks from the cloud to the edge. This allows moving ML computing close to the source of data, specifically to edge devices (computers) that are connected to cameras.

Performing on-device image recognition makes it possible to overcome the limitations of the cloud in terms of privacy, real-time performance, efficacy, robustness, and more. Hence, the use of Edge AI for computer vision makes it possible to scale image recognition applications in real-world scenarios.
The field of computer vision includes a set of main problems such as image classification, localization, image segmentation, and object detection. Among those, image classification can be considered the fundamental problem. It forms the basis for other computer vision problems.

Image classification applications are used in many areas, such as medical imaging, object identification in satellite images, traffic control systems, brake light detection, machine vision, and more. To find more real-world applications of image classification, check out our extensive list of AI vision applications.

Image classification is the task of categorizing and assigning labels to groups of pixels or vectors within an image dependent on particular rules. The categorization law can be applied through one or multiple spectral or textural characterizations.

The Machine Learning algorithm that is extremely good at classifying things is called a Convolutional Neural Network. [58]

4.2.2. PATTERN RECOGNITION

Pattern recognition analyzes incoming data and tries to identify patterns. While explorative pattern recognition aims to identify data patterns in general, descriptive pattern recognition starts by categorizing the detected patterns. Hence, pattern recognition deals with both of these scenarios, and different pattern recognition methods are applied depending on the use case and form of data.

Consequently, pattern recognition is not *one* technique but rather a broad collection of often loosely related knowledge and techniques. Pattern recognition capability is often a prerequisite for intelligent systems.

The data inputs for pattern recognition can be words or texts, images, or audio files. Hence, pattern recognition is broader compared to computer vision which focuses on image recognition.

Automatic and machine-based recognition, description, classification, and grouping of patterns are important problems in a variety of engineering and scientific disciplines, including biology, psychology, medicine, marketing, computer vision, and artificial intelligence. [59]

Historically, the two major approaches to pattern recognition are

- Statistical Pattern Recognition (or decision-theoretic) and
- Syntactic Pattern Recognition (or structural).

The third major approach is based on the technology of artificial neural networks (ANN), named

- Neural Pattern Recognition.

No single technology is always the optimal solution for a given pattern recognition problem. All three or hybrid methods are often considered to solve a given pattern recognition problem. [59]

4.2.3. IMAGE RECOGNITION

Image Recognition is the task of identifying objects of interest within an image and recognizing which category they belong to. Image recognition, photo recognition, and picture recognition are terms that are used interchangeably.

When we visually see an object or scene, we automatically identify objects as different instances and associate them with individual definitions. However, visual recognition is a highly complex task for machines to perform.

Image recognition with artificial intelligence is a long-standing research problem in the computer vision field. While different methods evolved, the common goal of image recognition is the classification of detected objects into different categories. Therefore, it is also called object recognition.

In past years, machine learning, in particular deep learning technology, has achieved big successes in many computer vision and image understanding tasks. Hence, deep learning image recognition methods achieve the best results in terms of performance (computed frames per second/FPS) and flexibility. Later in this article, we will cover the best performing deep learning algorithms and AI models for image recognition. [59]

How does Image Recognition work?

- Using traditional Computer Vision

The conventional computer vision approach to image recognition is a sequence of image filtering, segmentation, feature extraction, and rule-based classification. However, the traditional computer vision approach requires a high level of expertise, a lot of engineering time, and contains many parameters that need to be manually determined, while the portability to other tasks is pretty limited.

- Using Machine Learning and Deep Learning

Image recognition with machine learning, on the other hand, uses algorithms to learn hidden knowledge from a dataset of good and bad samples (Supervised Learning). The most popular machine learning method is deep learning, where multiple hidden layers are used in a model. The introduction of deep learning in combination with powerful AI hardware and GPUs enabled great breakthroughs in the field of image recognition. With deep learning, image classification and face recognition algorithms achieve above human-level performance and real-time object detection.

4.3. MACHINE LEARNING IMAGE RECOGNITION MODELS

1. SUPPORT VECTOR MACHINE

Support vector machine (SVM) is a machine learning method developed in the mid-1990s based on statistical learning theory. SVM classifier is currently the more popular classifier. This paper presents a boundary detection technique for retaining the potential support vector. Seeking structural risk minimization of the SVM improves the learning generalization ability and achieves the minimization of empirical risk and confidence range in the case of a small statistical sample size and it can also obtain the desired good statistically. [60]

SVMs work by making histograms of images containing the target objects and also of images that don't. The algorithm then takes the test picture and compares the trained histogram values with the ones of various parts of the picture to check for matches. [59]

Support vector machine is to map the low-dimensional space point to 02the high-dimensional space so that they become linearly separable. Then use the principles of linear division to determine the c1assification of the border. In the high-dimensional space, it is a linear division, while in the original data space, it is a non-linear division. [60]

Support vector machines are a set of supervised learning methods used for classification, regression, an outlier's detection. All of these are common tasks in machine learning. You can use them to detect cancerous cells based on millions of images or you can use them to predict future driving routes with a well-fitted regression model. There are specific types of SVMs you can use for particular machine learning problems, like support vector regression (SVR) which is an extension of support vector classification (SVC). The main thing to keep in mind here is that these are just math equations tuned to give you the most accurate answer possible as quickly as possible. SVMs are different from other classification algorithms because of the way they choose the decision boundary that maximizes the distance from the nearest data points of all the classes. The decision boundary created by SVMs is called the maximum margin classifier or the maximum margin hyperplane.

How does an SVM work?

A simple linear SVM classifier works by making a straight line between two classes. That means all of the data points on one side of the line will represent a category and the data points on the other side of the line will be put into a different category. This means there can be an infinite number of lines to choose from.

What makes the linear SVM algorithm better than some of the other algorithms, like k-nearest neighbors, is that it chooses the best line to classify your data points. It chooses the line that separates the data and is the furthest away from the closest data points as possible.

A 2-D example helps to make sense of the entire machine learning jargon. You have some data points on a grid. You're trying to separate these data points by the category they should fit in, but you don't want to have any data in the wrong category. That means you're trying to find the line between the two closest points that keeps the other data points separated. So the two closest data points give you the support vectors you'll use to find that line. That line is called the decision boundary.

Types of SVMs

There are two different types of SVMs, each used for different things:

Simple SVM: Typically used for linear regression and classification problems.

Kernel SVM: Has more flexibility for non-linear data because you can add more features to fit a hyperplane instead of a two-dimensional space.

We did simple SVM for linear regression and classification.

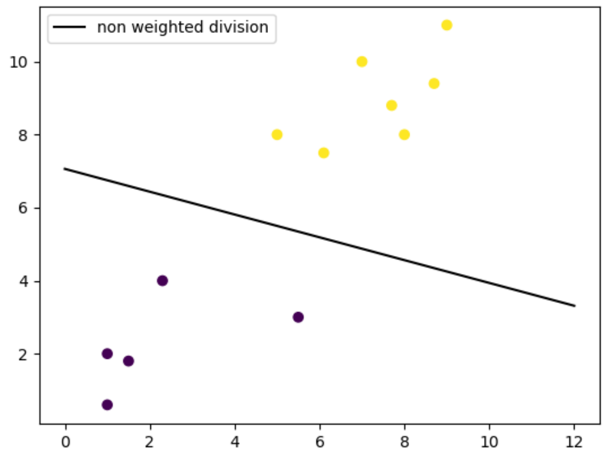


Figure 18

3. RANDOM FOREST

Random Forest (RF) is one of the many machine learning algorithms used for supervised learning, this means learning from labeled data and making predictions based on the learned patterns. RF can be used for both classification and regression tasks.

Decision tree

RF is based on decision trees. In machine learning, decision trees are a technique for creating predictive models. They are called decision trees because the prediction follows several branches of "if… then…" decision splits - similar to the branches of a tree. If we imagine that we start with a sample, which we want to predict a class, we would start at the bottom of a tree and travel up the trunk until we come to the first split-off branch. This split can be thought of as a feature in machine learning, let's say it would be "age"; we would now decide on which branch to follow: "if our sample has an age bigger than 30, continue along the left branch, else continue along the right branch". This we would do until we come to the next branch and repeat the same decision process until there are no more branches before us. This endpoint is called a leaf and in decision, trees would represent the final result: a predicted class or value.

At each branch, the feature thresholds that best split the (remaining) samples locally are found. The most common metrics for defining the "best split" are Gini impurity and information gain for classification tasks and variance reduction for regression.

Single decision trees are very easy to visualize and understand because they follow a method of decision-making that is very similar to how we humans make decisions: with a chain of simple rules. However, they are not very robust, i.e. they don't generalize well to unseen samples. Here is where Random Forests come into play.

Ensemble learning

RF makes predictions by combining the results from many individual decision trees - so we call them a forest of decision trees. Because RF combines multiple models, it falls under the category of ensemble learning. Other ensemble learning methods are gradient boosting and stacked ensembles.

Combining decision trees

There are two main ways of combining the outputs of multiple decision trees into a random forest:

- 1. Bagging, which is also called Bootstrap aggregation (used in Random Forests)
- 2. Boosting (used in Gradient Boosting Machines)

Bagging works the following way: decision trees are trained on randomly sampled subsets of the data, while sampling is being done with replacement. Bagging is the default method used with Random Forests. A big advantage of bagging over individual trees is that it decreases the variance of the model. Individual trees are very prone to overfitting and are very sensitive to noise in the data. As long as our trees are not correlated, combining them with bagging will make them more robust without increasing the bias. The part about correlation is important, though! We remove (most of) the correlation by randomly sampling subsets of data and training the different decision trees on this subset instead of on the entire dataset. In addition to randomly sampling instances from our data, RF also uses feature bagging. With feature bagging, at each split in the decision tree, only a random subset of features is considered. This technique reduces correlation even more because it helps reduce the impact of very strong predictor variables. Boosting works similarly but with one major difference: the samples are weighted for sampling so that samples, which were predicted incorrectly get a higher weight and are therefore sampled more often. The idea behind this is that difficult cases should be emphasized during learning compared to easy cases. Because of this difference bagging can be easily paralleled, while boosting is performed sequentially. The final result of our model is calculated by averaging over all predictions from these sampled trees or by majority vote.

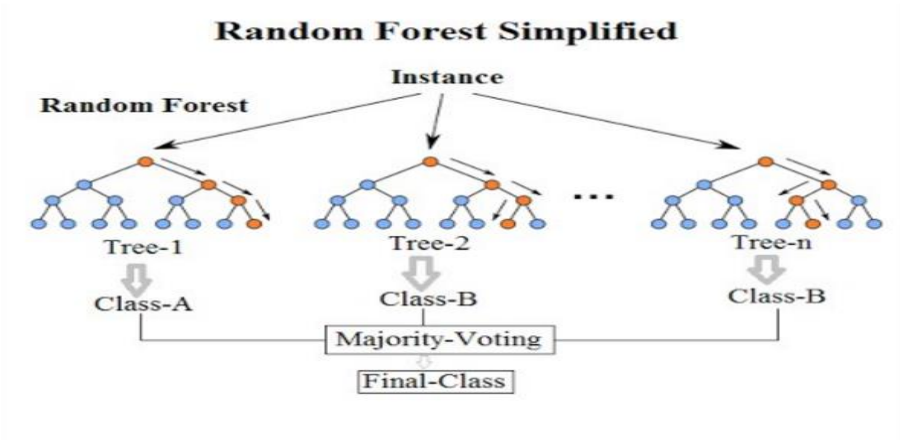


Figure 19

2. CONVOLUTIONAL NEURAL NETWORKS.

In the past few decades, Deep Learning has proved to be a very powerful tool because of its ability to handle large amounts of data. The interest to use hidden layers has surpassed traditional techniques, especially in pattern recognition. One of the most popular deep neural networks is the Convolutional Neural network. [32]

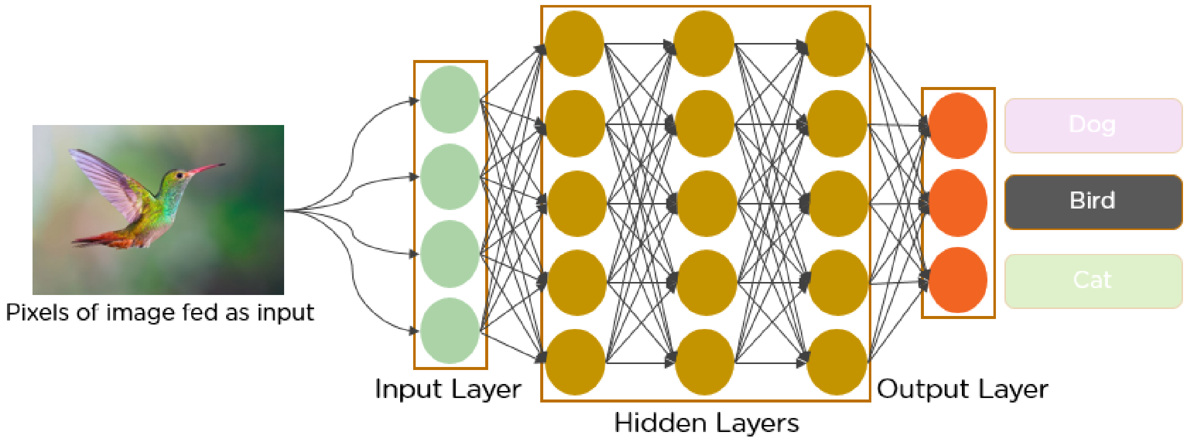


Figure 20

Since the 1950s, the early days of AI, researchers have struggled to make a system that can understand visual data. In the following years, this field came to be known as Computer Vision. In 2012, computer vision took a quantum leap when a group of researchers from the University of Toronto developed an AI model that surpassed the best image recognition algorithms and that too by a large margin.

The AI system, which became known as AlexNet (named after its main creator, Alex Krizhevsky), won the 2012 ImageNet computer vision contest with an amazing 85 percent accuracy. The runner-up scored a modest 74 percent on the test. At the heart of AlexNet were Convolutional Neural Networks a special type of neural network that roughly imitates human vision. Over the years CNNs have become a very important part of many Computer Vision applications and hence a part of any computer vision course online. So let's take a look at the workings of CNNs.

Background of CNNs

CNNs were first developed and used around the 1980s. The most that a CNN could do at that time was recognized written digits. It was mostly used in the postal sectors to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and also requires a lot of computing resources. This was a major drawback for CNNs at that period hence CNNs were only limited to the postal sectors and it they led to enter the world of machine learning. [32]

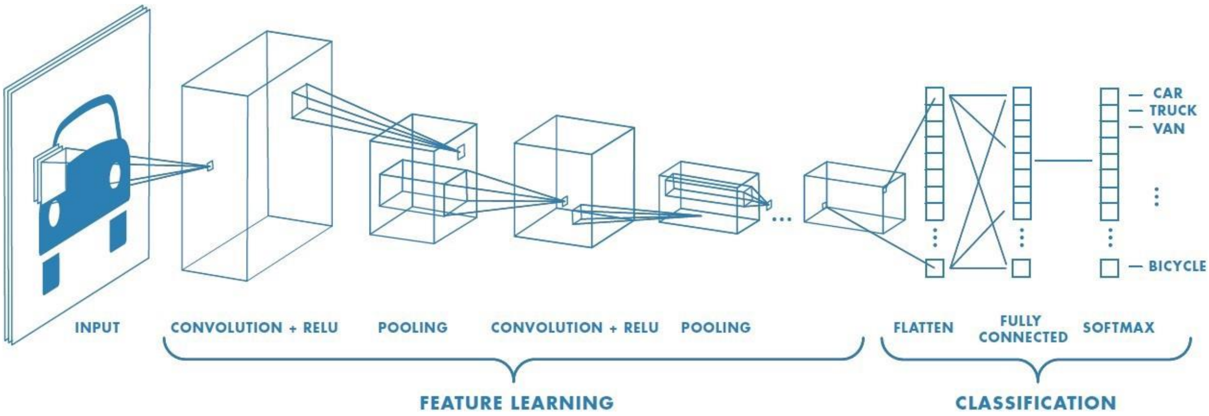


Figure 21

Artificial Intelligence has been witnessing a monumental growth in bridging the gap between the capabilities of humans and machines. Researchers and enthusiasts alike, work on numerous aspects of the field to make amazing things happen. One of many such areas is the domain of Computer Vision.

The agenda for this field is to enable machines to view the world as humans do, perceive similar layer, and even use the knowledge for a multitude of tasks such as Image & Video recognition, Image Analysis & Classification, Media Recreation, Recommendation Systems, Natural Language Processing, etc. The advancements in Computer Vision with Deep Learning have been constructed and perfected with time, primarily over one particular algorithm — a Convolutional Neural Network. [31]

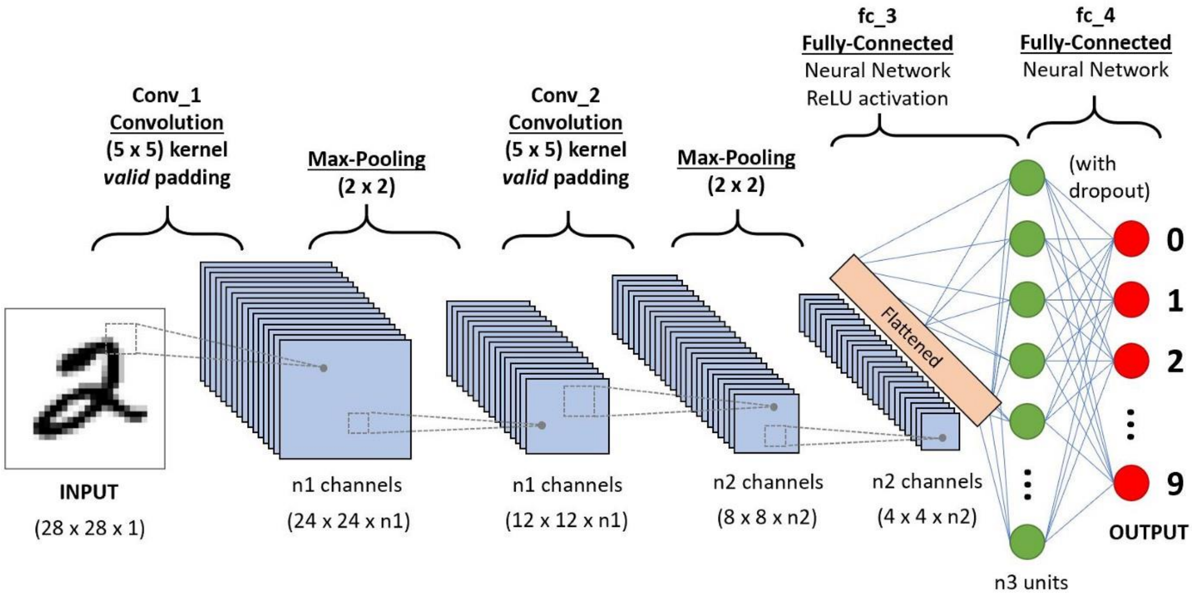


Figure 22

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm that can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image, and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the acanse filters/characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

Convolution

In mathematics, casually speaking, a mixture of two functions. In machine learning, a convolution mixes the convolutional filter and the input matrix in order to train weights.

The term "convolution" in machine learning is often a shorthand way of referring to either convolutional operation or convolutional layer.

Without convolutions, a machine learning algorithm would have to learn a separate weight for every cell in a large tensor. For example, a machine learning algorithm training on 2K x 2K images would be forced to find 4M separate weights. Thanks to convolutions, a machine learning algorithm only has to find weights for every cell in the convolutional filter, dramatically reducing the memory needed to train the model. When the convolutional filter is applied, it is simply replicated across cells such that each is multiplied by the filter.

Convolutional layer

A layer of a deep neural network in which a convolutional filter passes along an input matrix.

Convolutional neural network

A neural network in which at least one layer is a convolutional layer. A typical convolutional neural network consists of some combination of the following layers:

- convolutional layers
- pooling layers
- dense layers

Convolutional neural networks have had great success in certain kinds of problems, such as image recognition.

Why ConvNets over Feed-Forward Neural Nets?

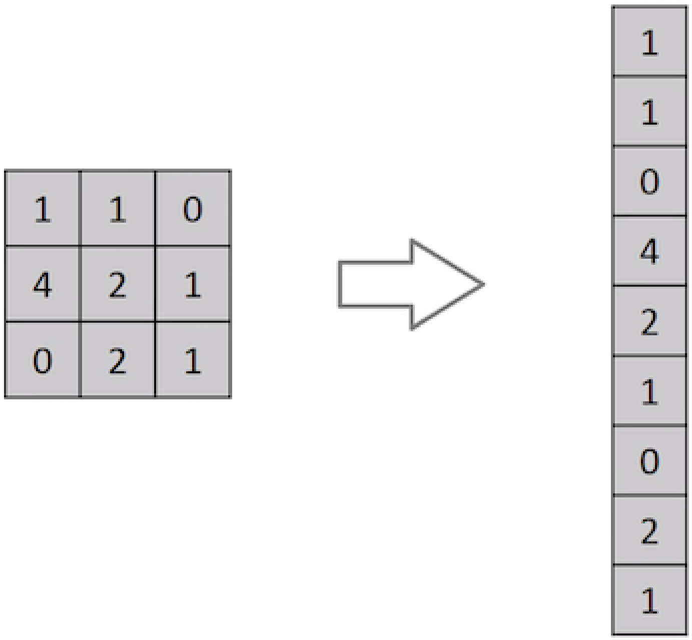


Figure 23

Flattening of a 3x3 image matrix into a 9x1 vector

An image is nothing but a matrix of pixel values. So why not just flatten the image (e.g. 3x3 image matrix into a 9x1 vector) and feed it to a Multi-Level Perceptron for classification purposes?

In cases of extremely basic binary images, the method might show an average precision score while performing prediction of classes but would have little to no accuracy when it comes to complex images having pixel dependencies throughout.

A ConvNet can successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and the reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

Input Image

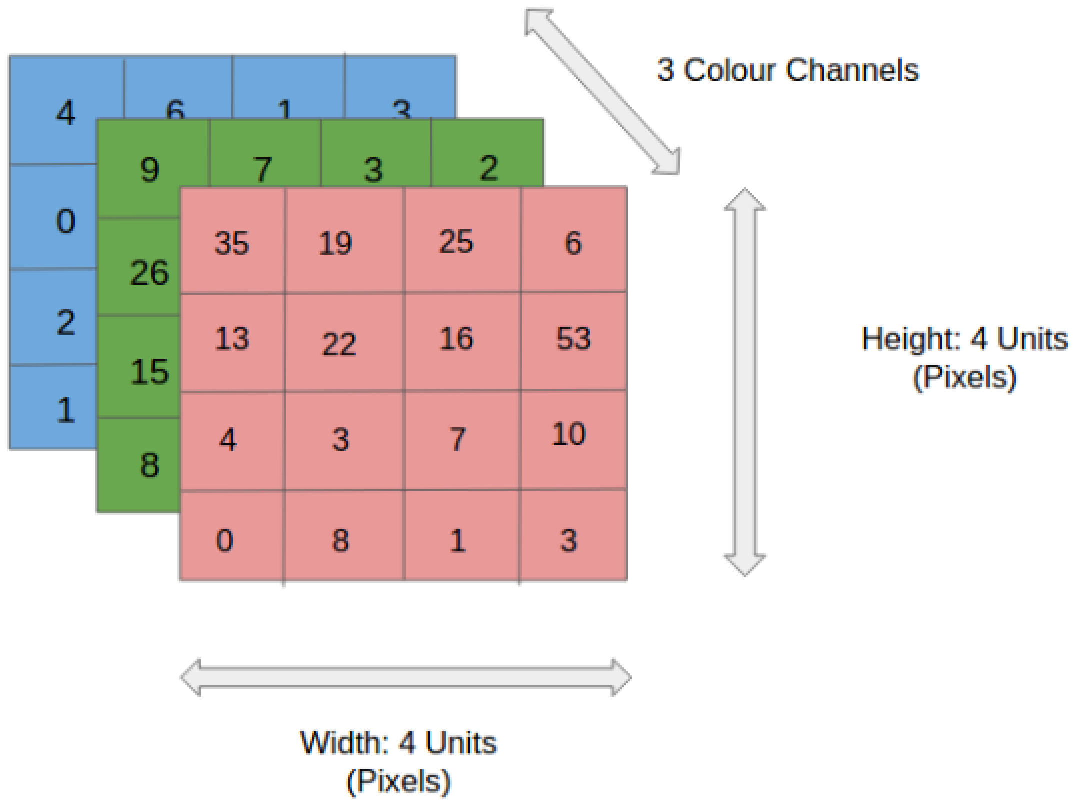


Figure 24

4x4x3 RGB Image

In the figure, we have an RGB image that has been separated by its three color planes — Red, Green, and Blue. There are a several color spaces in which images exist — Grayscale, RGB, HSV, CMYK, etc.

You can imagine how computationally intensive things would get once the images reach dimensions, say 8K (7680×4320). The role of the ConvNet is to reduce the images into a form that is easier to process, without losing features that are critical for getting a good prediction. This is important when we are to design an architecture that is not only good at learning features but also scalable to massive datasets.

For simplicity, let's stick with grayscale images as we try to understand how CNNs work.

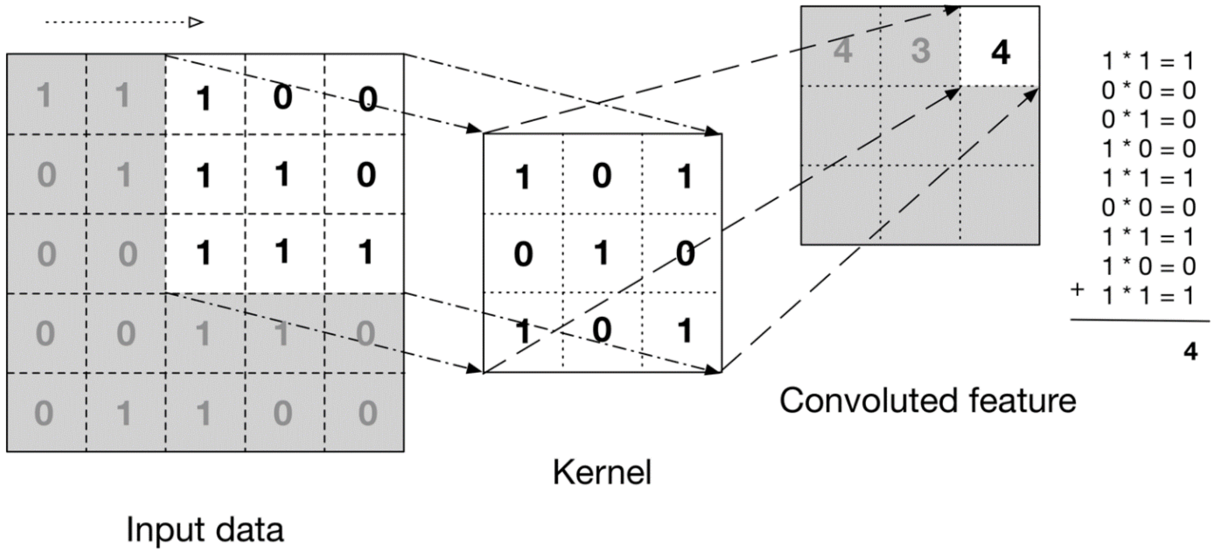


Figure 25

The above image shows what a convolution is. We take a filter/kernel (3×3 matrix) and apply it to the input image to get the convolved feature. This convolved feature is passed on to the next layer.

Convolution Layer — The Kernel

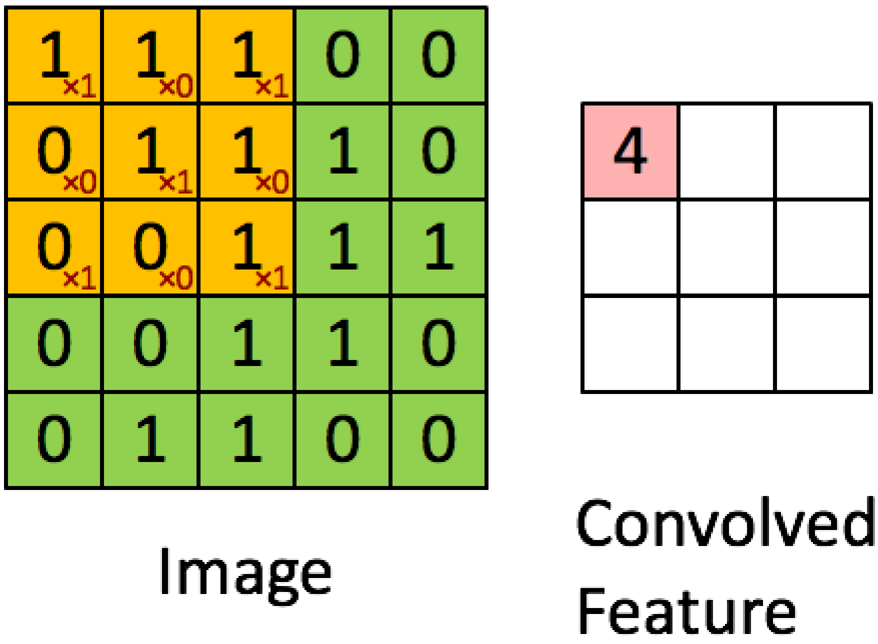


Figure 26

Convoluting a 5x5x1 image with a 3x3x1 kernel to get a 3x3x1 convolved feature

Image Dimensions = 5 (Height) x 5 (Breadth) x 1 (Number of channels, eg. RGB)

In the above demonstration, the green section resembles our 5x5x1 input image, I. The element involved in carrying out the convolution operation in the first part of a Convolutional Layer is called the Kernel/Filter, K, represented in the color yellow. We have selected K as a 3x3x1 matrix.

Kernel/Filter, K = 1 0 1

0 1 0

1 0 1

The Kernel shifts 9 times because of Stride Length = 1 (Non-Stride), every time performing a matrix multiplication operation between K and the portion P of the image over which the kernel is hovering.

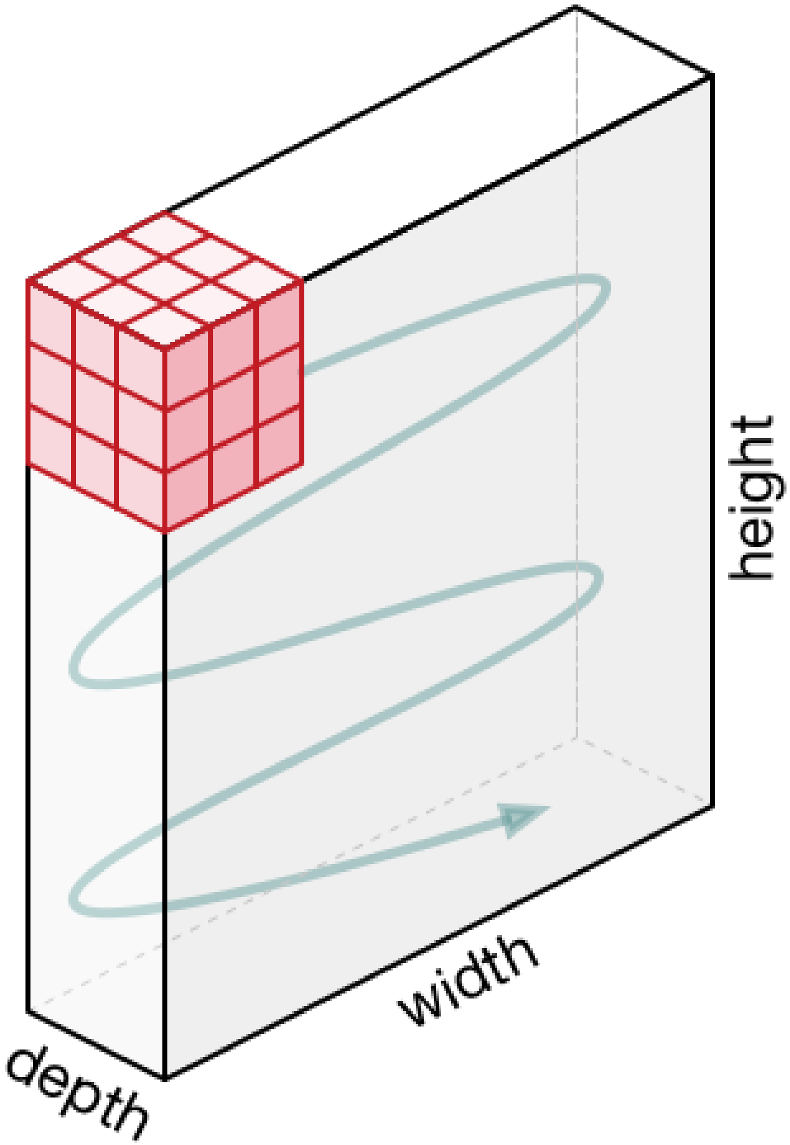


Figure 27

Movement of the Kernel

The filter moves to the right with a certain Stride Value till it parses the complete width. Moving on, it hops down to the beginning (left) of the image with the same Stride Value and repeats the process until the entire image is traversed.

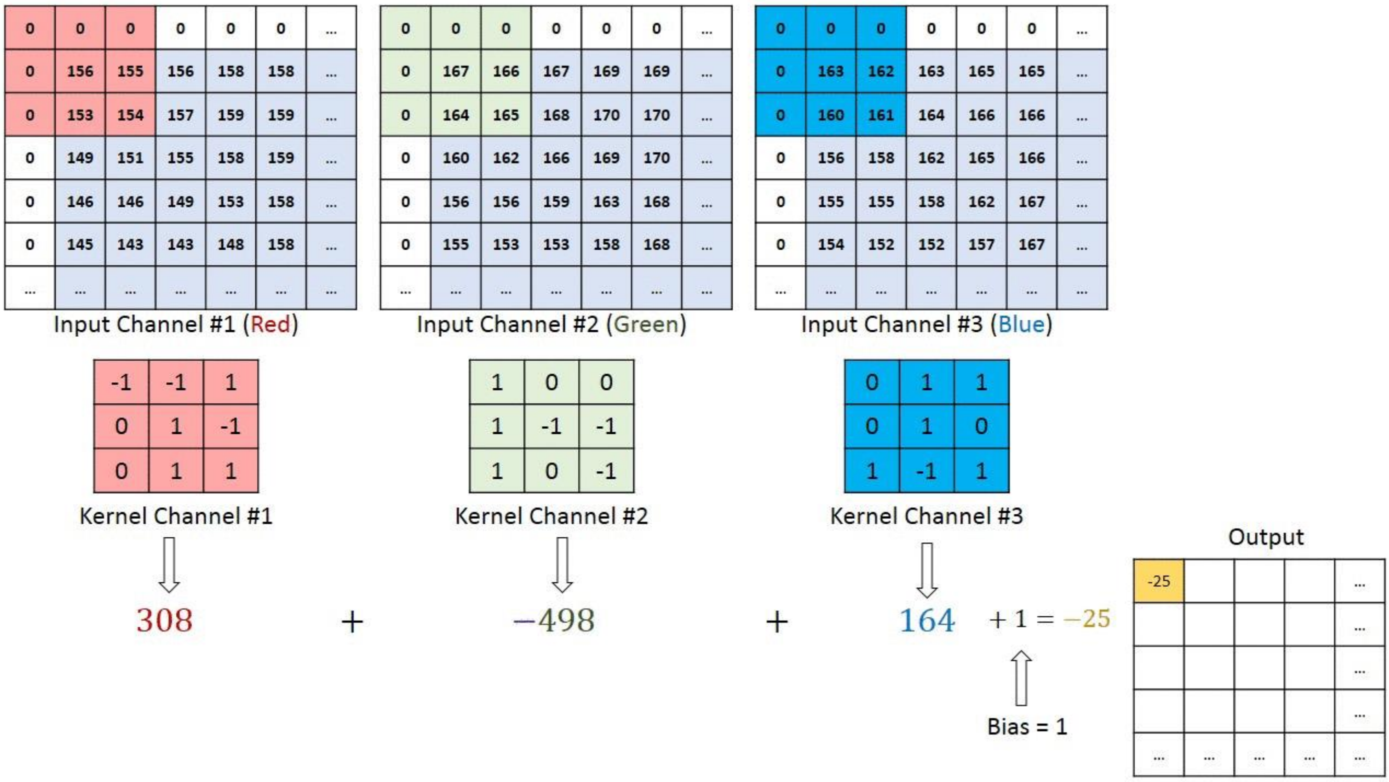


Figure 28

Convolution operation on an MxNx3 image matrix with a 3x3x3 Kernel.

In the case of images with multiple channels (e.g. RGB), the Kernel has the same depth as that of the input image. Matrix Multiplication is performed between Kn and in the stack ([K1, I1]; [K2, I2]; [K3, I3]) and all the results are summed with the bias to give us a squashed one-depth channel Convoluted Feature Output.

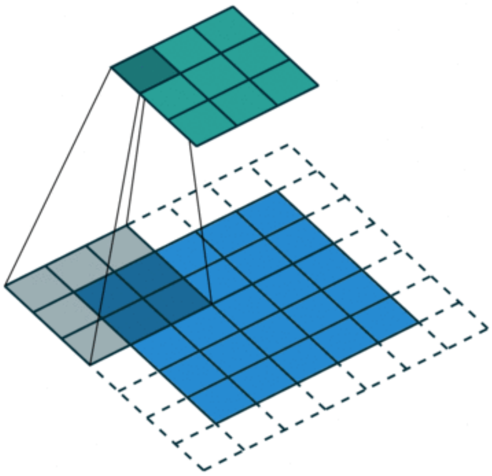


Figure 29

Convolution Operation with Stride Length = 2

The objective of the Convolution Operation is to extract the high-level features such as edges, from the input image. ConvNets need not be limited to only one Convolutional Layer. Conventionally, the first ConvLayer is responsible for capturing the Low-Level features such as edges, color, gradient orientation, etc. With added layers, the architecture adapts to the High-Level features as well, giving us a network that has a wholesome understanding of images in the dataset, similar to how we would.

There are two types of results to the operation — one in which the convolved feature is reduced in dimensionality as compared to the input, and the other in which the dimensionality is either increased or remains the same. This is done by applying Valid Padding in the case of the former, or the Same Padding in the case of the latter.

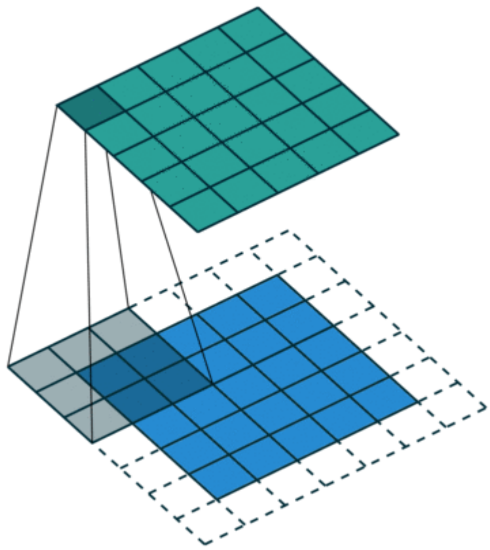


Figure 30

SAME padding: 5x5x1 image is padded with 0s to create a 6x6x1 image

When we augment the 5x5x1 image into a 6x6x1 image and then apply the 3x3x1 kernel over it, we find that the convolved matrix turns out to be of dimensions 5x5x1. Hence the name — same padding.

On the other hand, if we perform the same operation without padding, we are presented with a matrix that has dimensions of the Kernel (3x3x1) itself — Valid Padding.

Pooling Layer

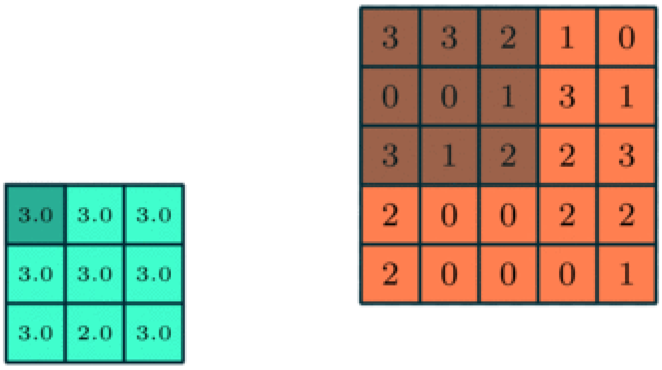


Figure 31

3x3 pooling over 5x5 convolved feature

Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training the model. There are two types of Pooling: Max Pooling and Average Pooling. Max Pooling returns the maximum value from the portion of the image covered by the Kernel. On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel.

Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that Max Pooling performs a lot better than Average Pooling.

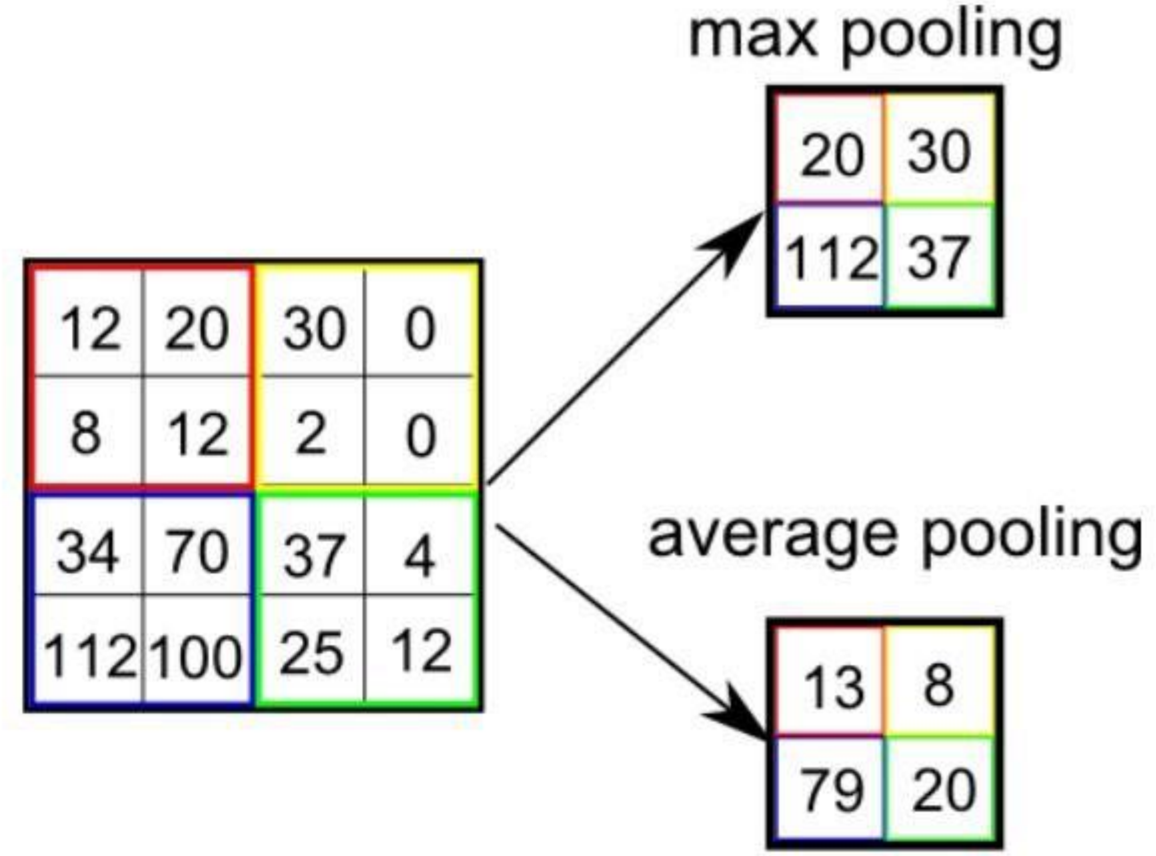


Figure 32

Types of Pooling

The Convolutional Layer and the Pooling Layer, together form the i-th layer of a Convolutional Neural Network. Depending on the complexities in the images, the number of such layers may be increased for capturing low-level details even further, but at the cost of more computational power.

After going through the above process, we have successfully enabled the model to understand the features. Moving on, we are going to flatten the final output and feed it to a regular Neural Network for classification purposes.

Classification — Fully Connected Layer (FC Layer)

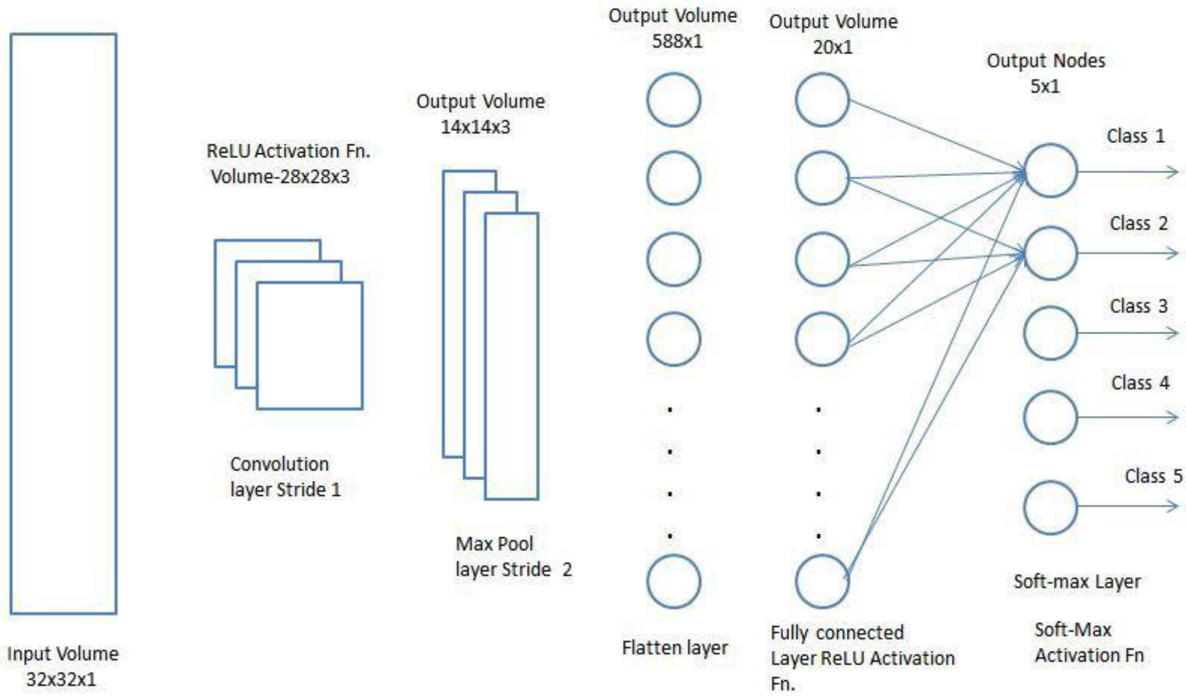


Figure 33

Adding a Fully-Connected layer is a (usually) cheap way of learning non-linear combinations of the high-level features as represented by the output of the convolutional layer. The Fully-Connected layer is learning a possibly non-linear function in that space.

Now that we have converted our input image into a suitable form for our Multi-Level Perceptron, we shall flatten the image into a column vector. The flattened output is fed to a feed-forward neural network and backpropagation is applied to every iteration of training. Over a series of epochs, the model can distinguish between dominating and certain low-level features in images and classifies them using the Softmax Classification technique. [31]

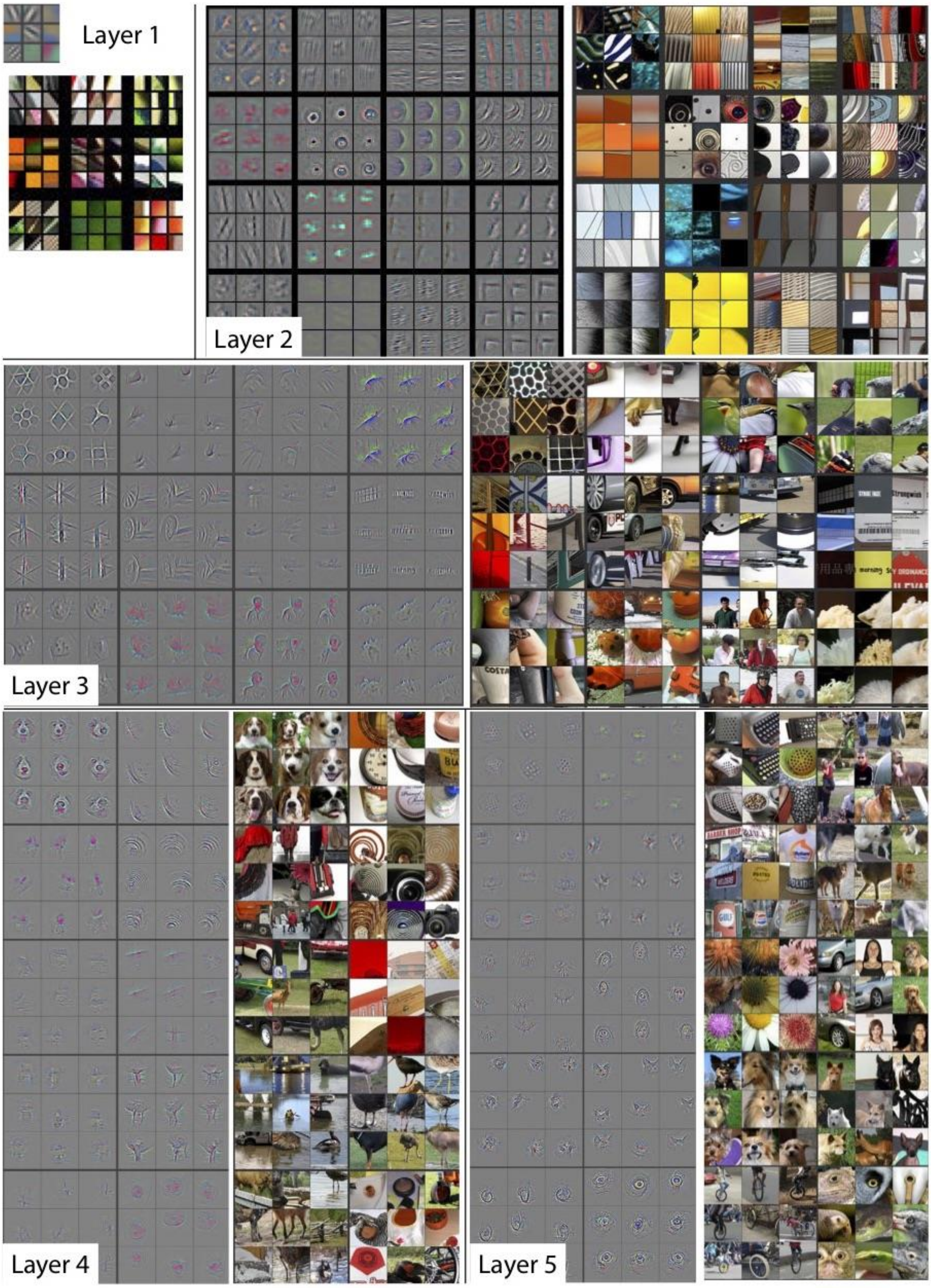


Figure 34

Based on the activation map of the final convolution layer, the classification layer outputs a set of confidence scores (values between 0 and 1) that specify how likely the image is to belong to a "class." For instance, if you have a ConvNet that detects cats, dogs, and horses, the output of the final layer is the possibility that the input image contains any of those animals.

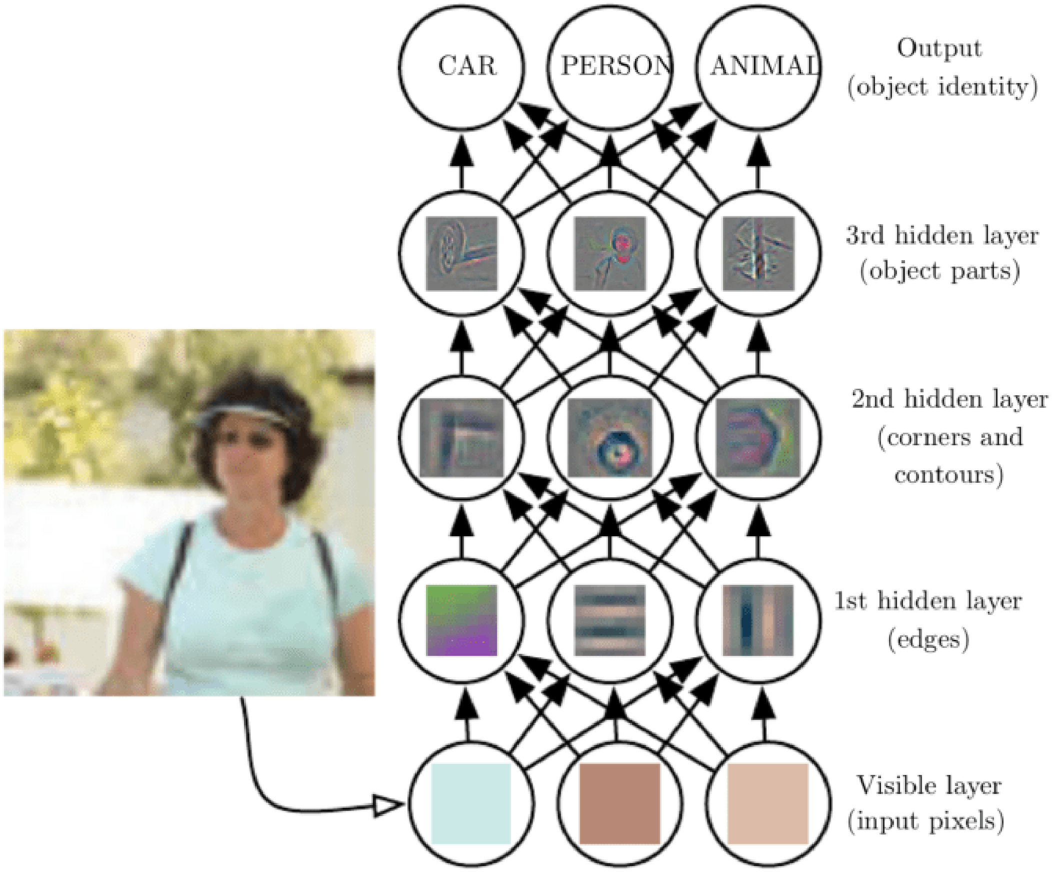


Figure 35

Limitations

Despite the power and resource complexity of CNNs, they provide in-depth results. At the root of it all, it is just recognizing patterns and details that are so minute and inconspicuous that it goes unnoticed by the human eye. But when it comes to understanding the contents of an image it fails.

For example, when we pass the below image to a CNN it detects a person in their mid-30s and a child probably around 10 years. But when we look at the same image we start thinking of multiple different scenarios. Maybe it's a father and son day out, a picnic or maybe they are camping. Maybe it is a school ground and the child scored a goal and his dad is happy so he lifts him.

These limitations are more than evident when it comes to practical applications. For example, CNN was widely used to moderate content on social media. But despite the vast resources of images and videos that they were trained on it still isn't able to completely block and remove inappropriate content.

Several studies have shown that CNNs trained on ImageNet and other popular datasets fail to detect objects when they see them under different lighting conditions and from new angles.

Despite the limits of convolutional neural networks, however, there's no denying that they have caused a revolution in artificial intelligence. Today, CNNs are used in many computer vision applications such as facial recognition, image search, and editing, augmented reality, and more. As advances in convolutional neural networks show, our achievements are remarkable and useful, but we are still very far from replicating the key components of human intelligence.[32]

TYPES OF CNN ARCHITECTURES

Among all of the state-of-the-art CNN architectures, we used ResNet50, MobileNet, and VGG16 as these models are rather updated and showed excellent results on the ImageNet dataset. [17]

1. RESIDUAL NETWORK

Residual Network was introduced in 2015 by the Microsoft Research team [35]. It is one of the most impressive and widely used networks all over the world since this network was designed to use hundreds of Conv layers without losing effectiveness. The performance of the convolutional neural network drops when a large number of Conv layers are stacked together. During weights update in the backpropagation process, the repeated multiplication results in a smaller gradient and this continues to become small and weights update very slowly which is referred to as the vanishing gradient problem. [35] Due to this, the performance drops rapidly. ResNet model solved this problem very efficiently by using a 'shortcut path' in parallel to the main path. The main purpose behind this connection is to ensure the information flow from one layer to another layer within a residual block by skipping some intermediate layers. In that way, more information is added on a deeper layer and this prevents the vanishing gradient problem. The filter for convolution is mostly 3×3 size and the ResNet-50 network has 14 residual blocks. The global average pooling layer is used on top of the final residual block followed by a dense layer with softmax activation. [17]

3. RESNET 50

Residual networks (ResNets) are deep convolutional networks that help to solve the problem of vanishing gradients in deep networks by introducing an "identity shortcut connection" that skips the block of convolution layers. ResNets use shortcut connections that form an alternate path for the gradient to be directly back-propagated to previous layers. ResNet50 architecture has 5 stages with convolution and shortcut connections that skip over a block of 3 layers. ResNet50 has 48 convolution layers, one max pool layer, and one average pool layer. ResNet50V2 architecture is a modified version of ResNet50 and has about 25 million trainable parameters. ResNet50 takes as input, the images of size 224 × 224 × 3 and produces a feature vector of size 2048.

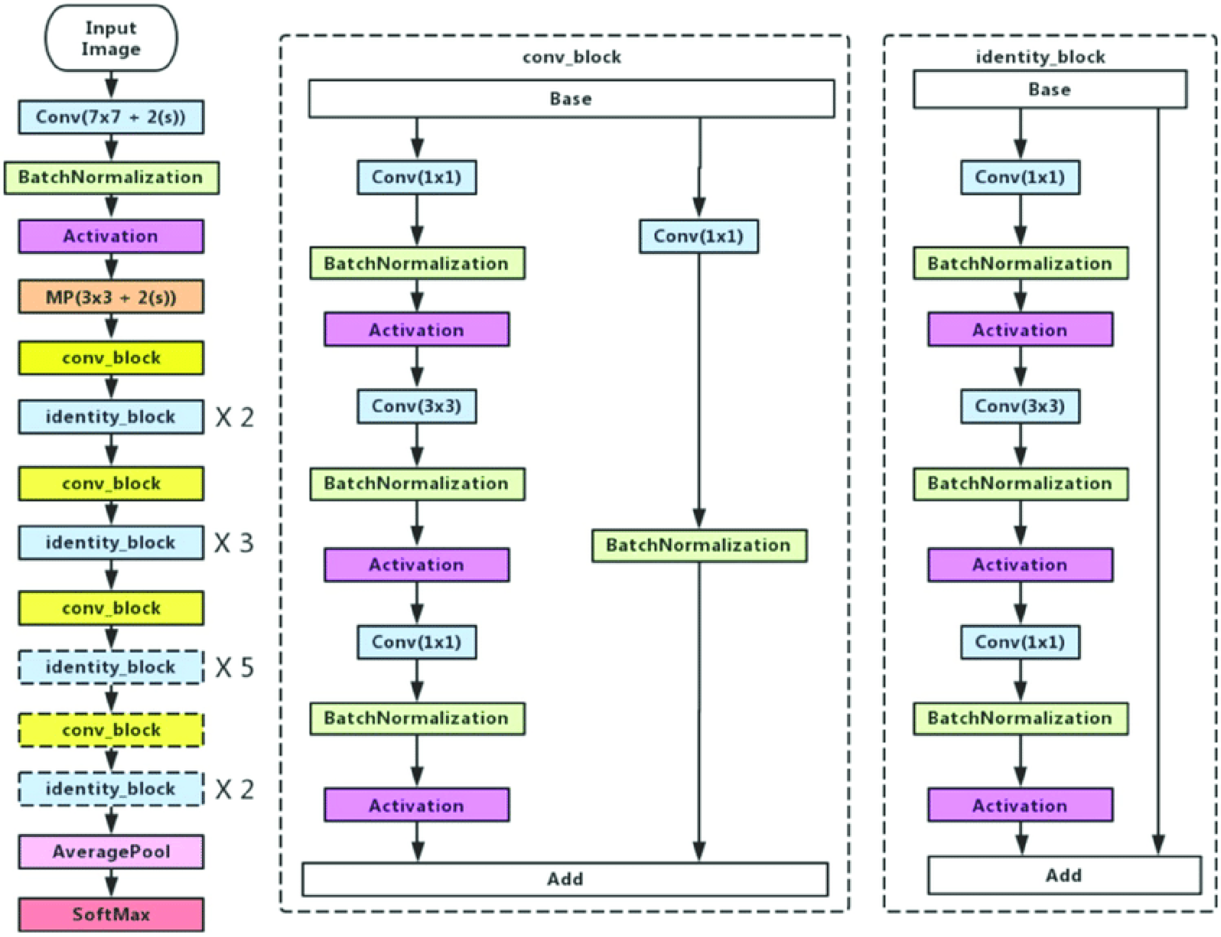


Figure 36

3.MobileNet

MobileNet was proposed by Howard et al. (2017). It is designed for mobile devices and integrated vision systems. This network is composed of 28 layers, including 13 depthwise convolutions and 13 pointwise convolutions. The main purpose of depthwise convolution modules is to reduce the dimension of the network. In MobileNet, all layers are followed by batch normalization and ReLU, and the final layer is a fully connected layer that feeds to the softmax. The MobileNet network introduces two hyperparameters to realize a compromise between the response period, storage space, and precision. These hyperparameters are the width multiplier and the resolution multiplier. The role of the width multiplier is to reduce the number of feature maps in convolution layers. In contrast, the resolution multiplier reduces the computational cost by decreasing the input image's resolution, which automatically decreases the size of the convolution layers.

MobileNet uses Depthwise separable convolutions. It significantly reduces the number of parameters when compared to the network with regular convolutions with the same depth in the nets. This results in lightweight deep neural networks.

- A Depthwise separable convolution is made from two operations.
- 1. Depthwise convolution.
- 2. Pointwise convolution.

Depthwise Separable Convolution originated from the idea that a filter's depth and spatial dimension can be separated- thus, the name is separable. Let us take the example of the Sobel filter, used in image processing to detect edges.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Figure 37

You can separate the height and width dimensions of these filters. Gx filter can be viewed as a matrix product of [1 2 1] transpose with [-1 0 1].

We notice that the filter had disguised itself. It shows it had nine parameters, but it has 6. This has been possible because of the separation of its height and width dimensions.

The same idea applied to separate depth dimension from horizontal (width*height) gives us depth-wise separable convolution where we perform depth-wise convolution. After that, we use a 1*1 filter to cover the depth dimension.

One thing to notice is how many parameters are reduced by this convolution to output the same no. of channels. To produce one channel, we need 3*3*3 parameters to perform depth-wise convolution and 1*3 parameters to perform further convolution in-depth dimension.

Depthwise separable convolution is a Depthwise convolution followed by a pointwise convolution as follows:

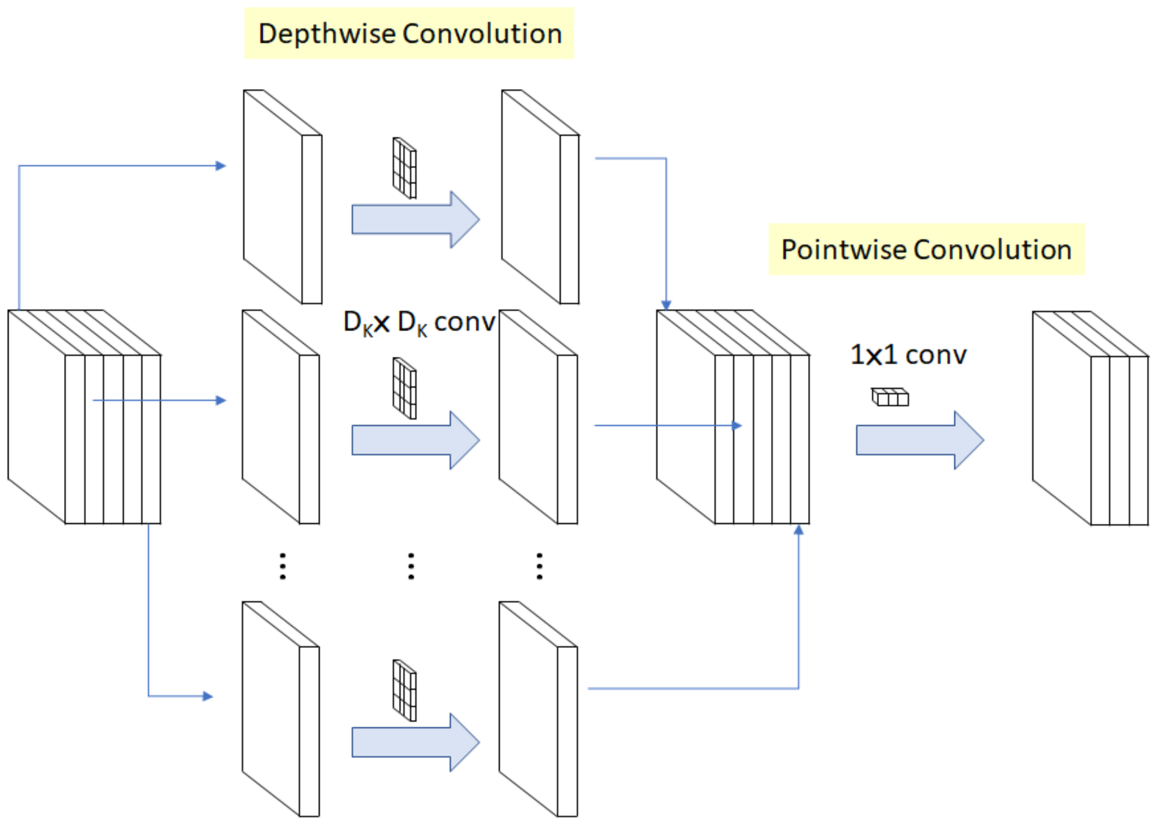


Figure 38

Pointwise convolution.

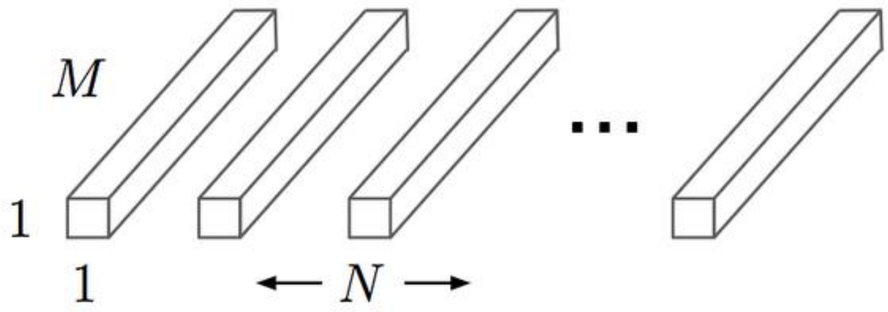


Figure 39

Convolution with a kernel size of 1x1 that simply combines the features created by the Depthwise convolution.

The main difference between MobileNet architecture and a traditional CNN instead of a single 3x3 convolution layer followed by the batch norm and ReLU. Mobile Nets split the convolution into a 3x3 depth-wise Conv and a 1x1 pointwise Conv, as shown in the figure.

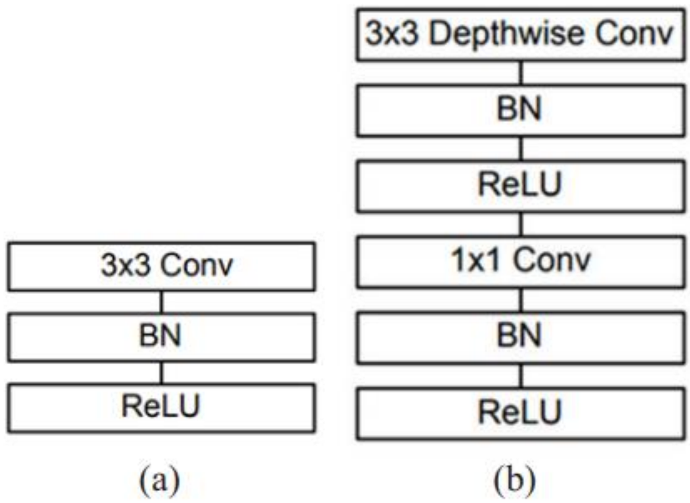


Figure 40

Fig. (a) Standard convolutional layer with batch normalization and ReLU.

Fig.(b) Depth-wise separable convolution with depth-wise and pointwise layers followed by batch normalization and ReLU.

MobileNet is a class of CNN that was open-sourced by Google, and therefore, this gives us an excellent starting point for training our classifiers that are insanely small and insanely fast.

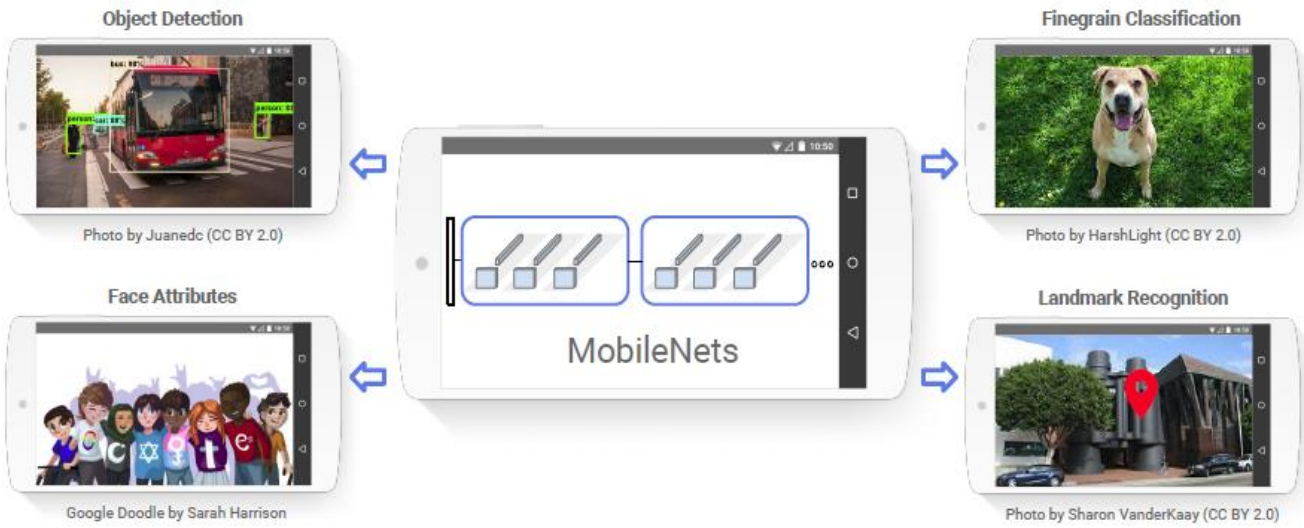


Figure 41

MobileNet Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
5× Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Table 8

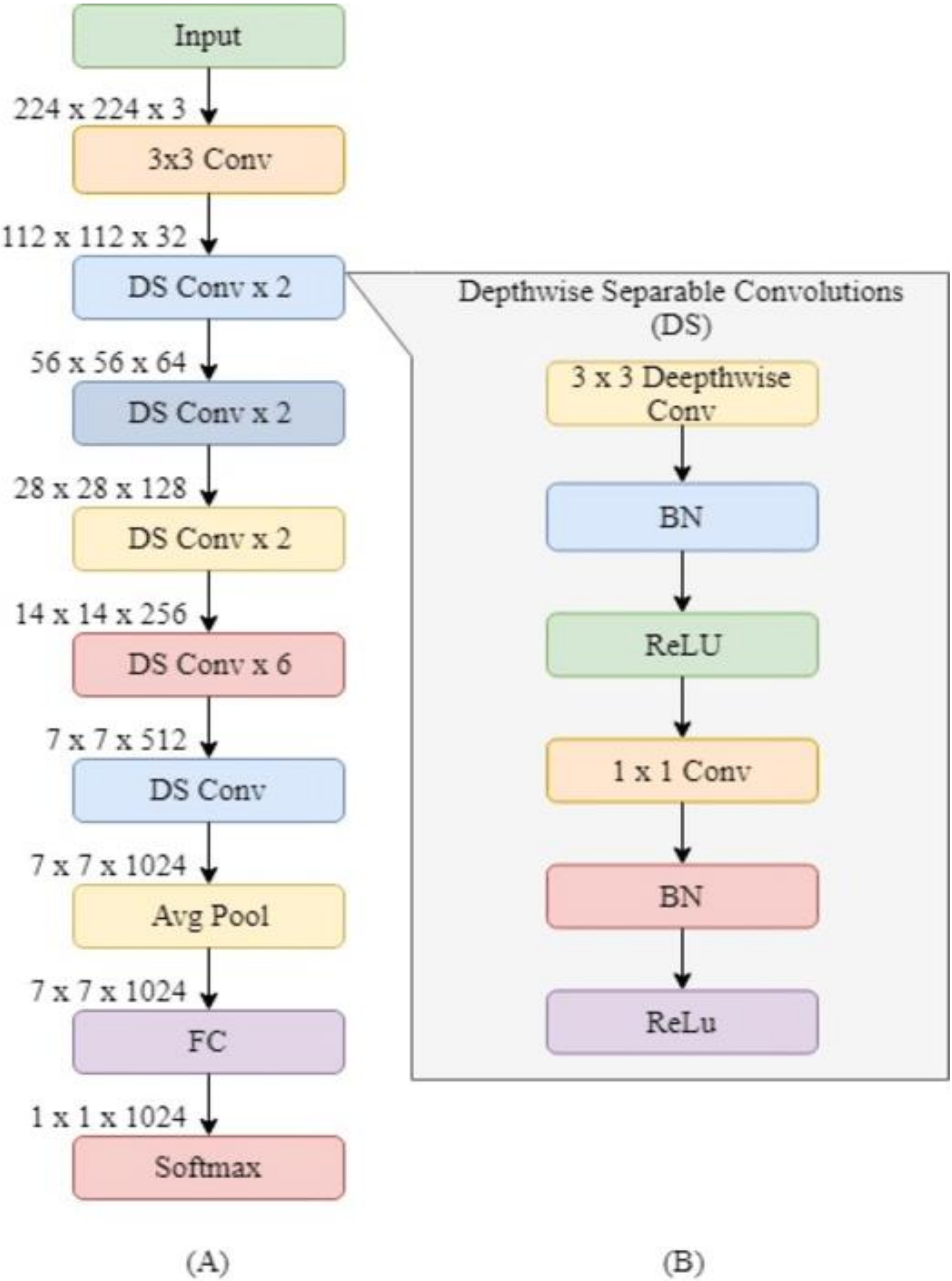

Input
224 x 224 x 3
3x3 Conv
112 x 112 x 32
DS Conv x 2
56 x 56 x 64
DS Conv x 2
28 x 28 x 128
DS Conv x 2
14 x 14 x 256
DS Conv x 6
7 x 7 x 512
DS Conv
7 x 7 x 1024
Avg Pool
7 x 7 x 1024
FC
1 x 1 x 1024
Softmax
Depthwise Separable Convolutions (DS)
3 x 3 Deepthwise Conv
BN
ReLU
1 x 1 Conv
BN
ReLu
(A)
(B)

Figure 42

4.VGG-16

VGG-16 is a convolutional neural network that is 16 layers deep. You can load a pre-trained version of the network trained on more than a million images from the ImageNet database [1]. The pre-trained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

The number 16 in the name VGG refers to the fact that it is 16 layers deep neural network (VGGnet). This means that VGG16 is a pretty extensive network and has a total of around 138 million parameters.

16 layers of VGG16

1. Convolution using 64 filters

2.Convolution using 64 filters + Max pooling

3. Convolution using 128 filters

4.Convolution using 128 filters + Max pooling

5. Convolution using 256 filters

6. Convolution using 256 filters

7.Convolution using 256 filters + Max pooling

8. Convolution using 512 filters

9. Convolution using 512 filters

10. Convolution using 512 filters + Max pooling

11. Convolution using 512 filters

12. Convolution using 512 filters

13. Convolution using 512 filters + Max pooling

14. Fully connected with 4096 nodes

15. Fully connected with 4096 nodes

16. Output layer with Softmax activation with 1000 nodes.

VGG-16 Architecture

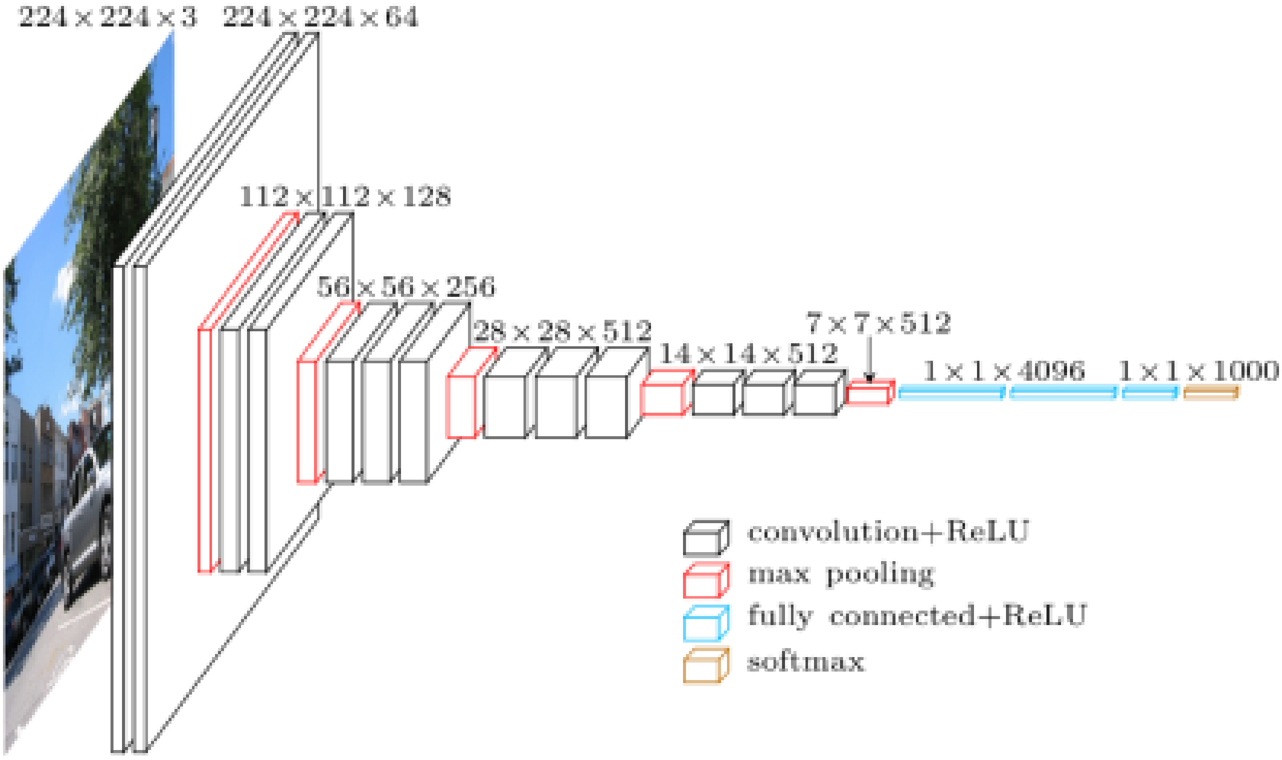


Figure 43

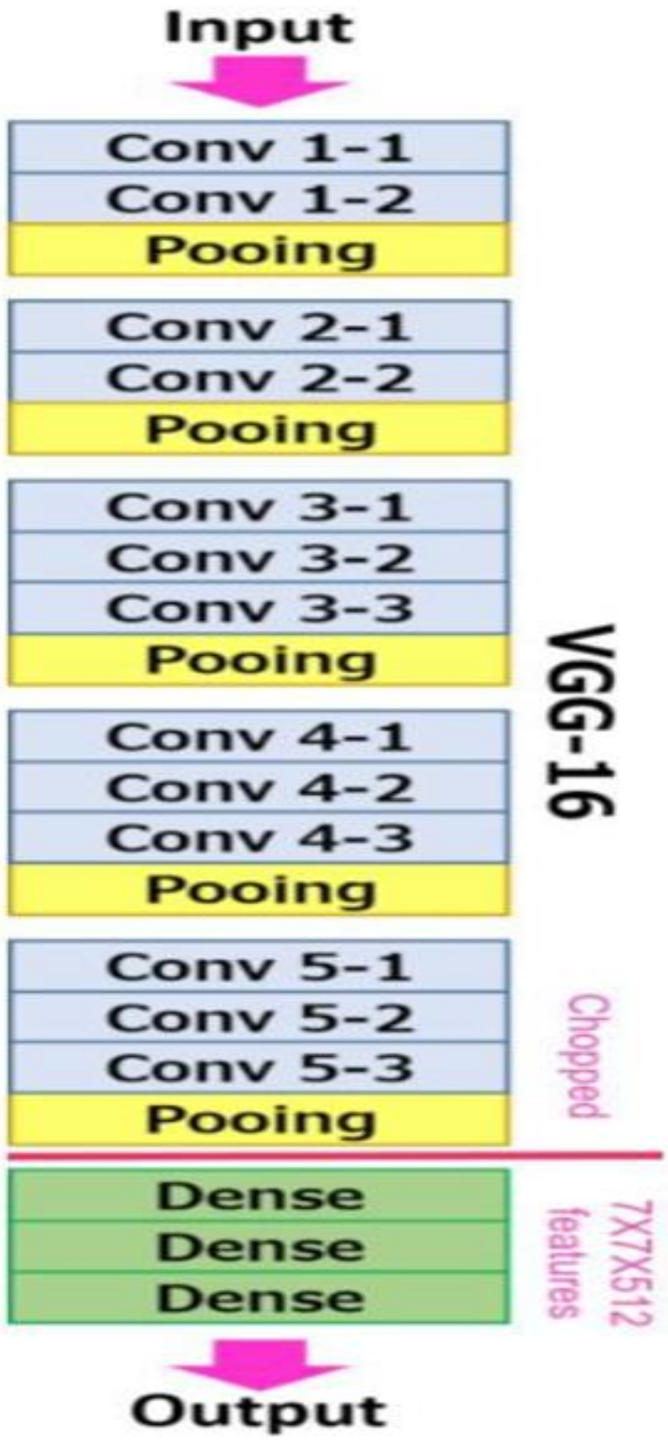

Input
Conv 1-1
Conv 1-2
Pooing
Conv 2-1
Conv 2-2
Pooing
Conv 3-1
Conv 3-2
Conv 3-3
Pooing
Conv 4-1
Conv 4-2
Conv 4-3
Pooing
Conv 5-1
Conv 5-2
Conv 5-3
Pooing
VGG-16
Chopped
Dense
Dense
Dense
7X7X512 features
Output

Figure 44

4. The proposed 4-CNN model

```
In [87]: model.summary()

Model: "sequential_6"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_18 (Conv2D)          (None, 112, 112, 64)      1792

 conv2d_19 (Conv2D)          (None, 112, 112, 16)      9232

 max_pooling2d_9 (MaxPooling  (None, 56, 56, 16)       0
 2D)

 dropout_5 (Dropout)         (None, 56, 56, 16)        0

 flatten_6 (Flatten)         (None, 50176)             0

 batch_normalization_6 (Batc  (None, 50176)            200704
 hNormalization)

 dense_6 (Dense)             (None, 8)                 401416

=================================================================
Total params: 613,144
Trainable params: 512,792
Non-trainable params: 100,352
_________________________________________________________________
```

Figure 45

We proposed a six-layer model to diagnose dermatoscopic images.

Our proposed architecture contains 2 convolutional layers (Conv2D_1, Conv2D_2), 1 max-pooling layer (Maxpool2D), 1 Dropout layer, 1 flatten layer, 1 batch normalization layer and 1 Dense layer.

Layers of Our Model

1. Convolution using 64 filters

2. Convolution using 16 filters + Max pooling

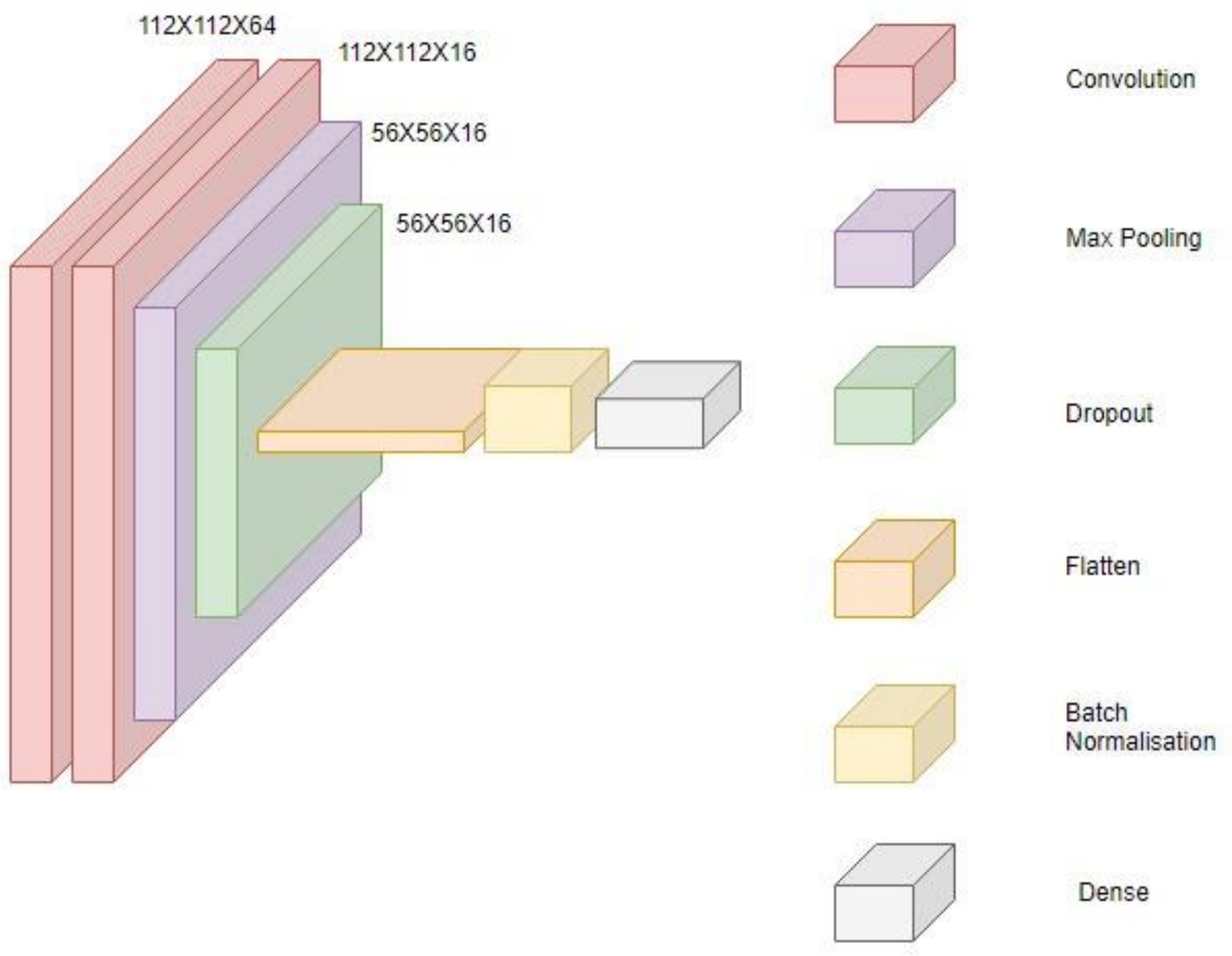
3. Dropout layer

4. Fully connected layer with 200704 nodes

5. Fully connected layer with 200704 nodes.

6. Output layer with activation sigmoid.

ARCHITECTURE



Software and hardware

We used Jupyter Notebook for running our python codes. The model was written utilizing the open-source Keras (ver. 2.2.4) library and the open-source TensorFlow (ver. 1.15.0) library as a backend for CNN models together with other scientific computing libraries such as NumPy and scikit-learn.

The architecture on which we run the experiments is Intel(R) Core(TM) i5-10400F CPU @ 2.90GHz, 16 GB RAM.

RESULT AND DISCUSSION

The proposed model, two classifiers: RF and SVM, and the three pre trained models: MobileNet, ResNet-50 and VGG-16, were evaluated using two different datasets: the HAM10000 and ISIC 2019. The lesion images present in HAM10000 are divided into 7 classes and images in ISIC 2019 are divided into 8 classes. Then these are further divided into training set and testing set. To understand the process of machine learning we started with two classifiers: RF and SVM. The below mentioned accuracy for 2 classes was just to test the system if it works properly. We didn't go directly for 7 classes because it takes more computational time. Many researchers have carried out binary classification such as melanoma and non-melanoma. But we worked on multi-class classification. The top1 accuracies of all these models are given in the table below. We compared the performance accuracies of the proposed model and that of state-of-the-art approaches for the skin lesion classification task. We split the two datasets into training and testing parts with same split ratio i.e. 80:20. Some researchers used a different dataset and a different proportion of the testing data from the whole dataset. For example, some use 30% of the dataset as the testing data, while some use only 10% for the testing data on the same dataset HAM10000 and ISIC2019. We applied augmentation for better accuracy. Total number of images in ISIC 2019 dataset are 25,331 and Total number of images after augmentation are 30,320. Of the total set of 30,320 dermatoscopic images, 25251images were used for training and 5069 images for testing. Total number of images in HAM10000 dataset are 10,015 and total number of images after augmentation are11,950. Of the total set of 11,950 dermatoscopic images, 9945 images were used for training and 2005 images for testing. Results are mentioned below in tabular form in table 4 and table 5. Training and validation graphs are also shown below.

1. FOR THE HAM10000 DATASET

Table 4

SR NO	ALGORITHM USED	CLASSES	ACCURACY	IMPROVED ACCURACY
1	SVM	2	63.43%	63.43%
2	SVM	7	66.30%	66.30%
3	RF	2	69.07%	69.07%
4	RF	7	69.74%	69.74%
5	MobileNet	7	91.02 %	92.42%
6	VGG-16	7	75.81 %	87.70 %
7	ResNet-50	7	67.23%	71.60%
8	Proposed model	2	98.44%	98.44%
9	Proposed model	7	85.71 %	97.04%

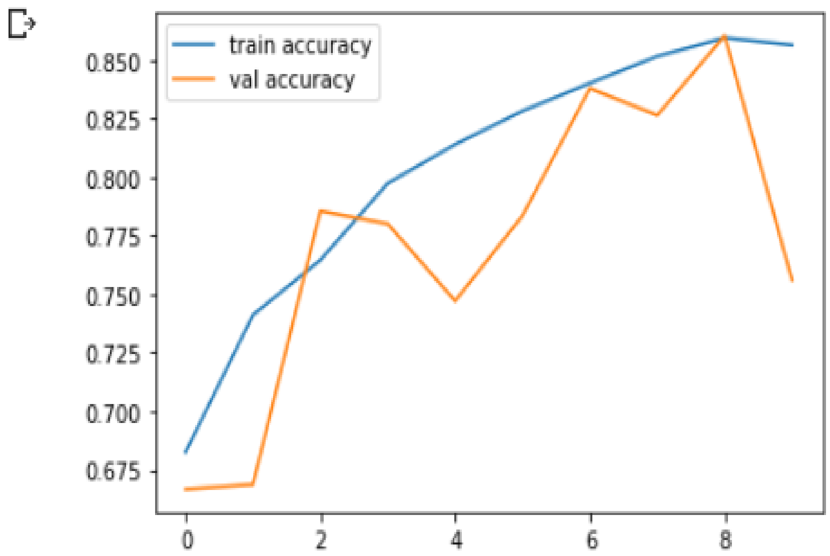
2. FOR ISIC 2019 DATASET

Table 5

SR NO	ALGORITHM USED	CLASSES	ACCURACY	IMPROVED ACCURACY
1	MobileNet	7	85.54%	93.63%
2	VGG-16	7	70.03%	70.03%
3	ResNet-50	7	55.10%	61%
4	Proposed model	7	84.21%	93.52%

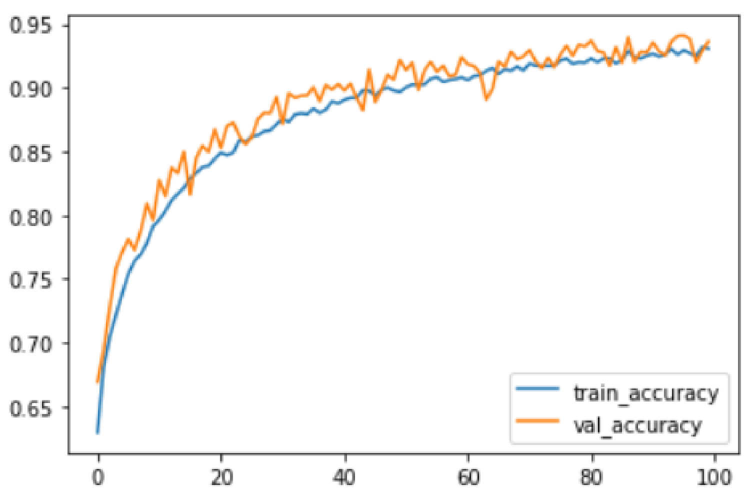
GRAPHICAL REPRESENTATION OF TRAINING AND VALIDATION ACCURACY

Mobilenet HAM10000

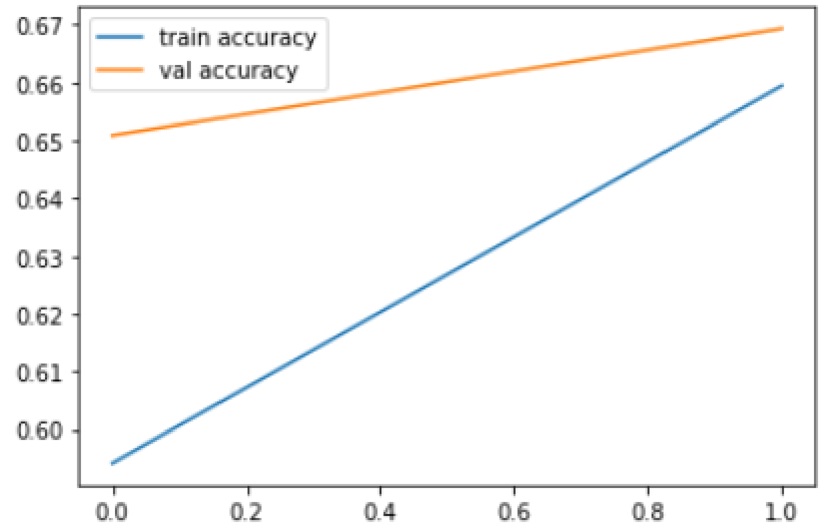


<Figure size 432x288 with 0 Axes>

Mobilenet ISIC 2019

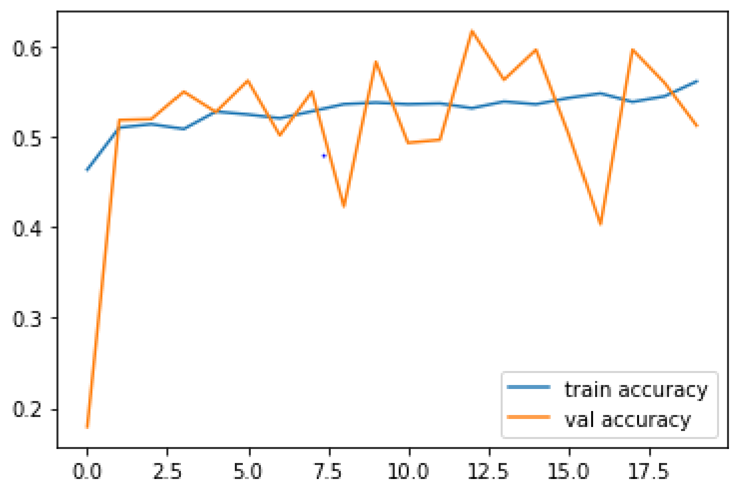


ResNet50 HAM10000



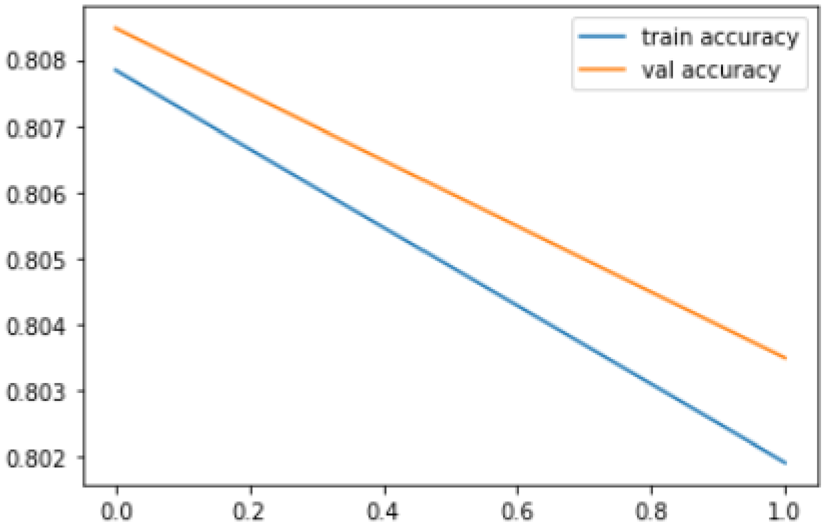
<Figure size 432x288 with 0 Axes>

ResNet50 ISIC 2019



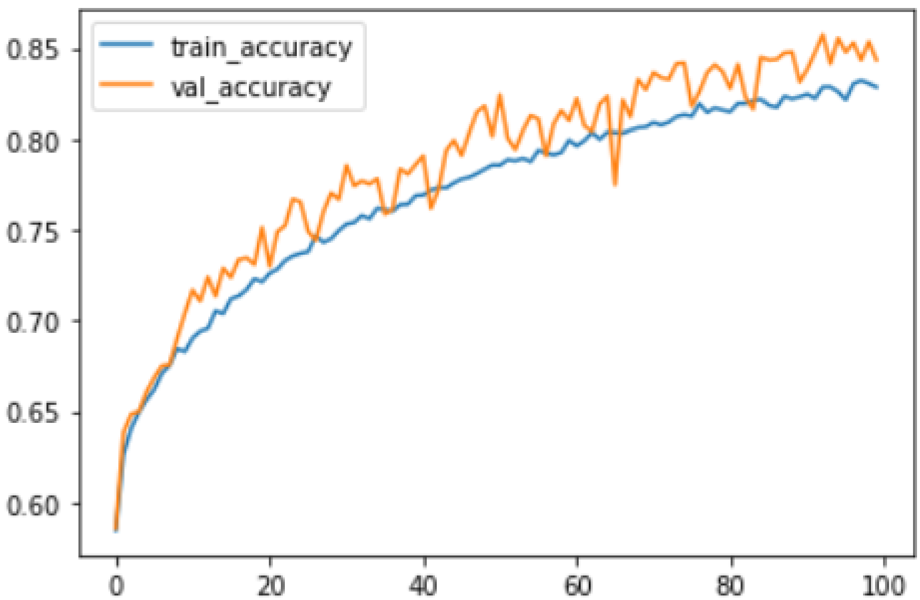
<Figure size 432x288 with 0 Axes>

VGG16 HAM10000

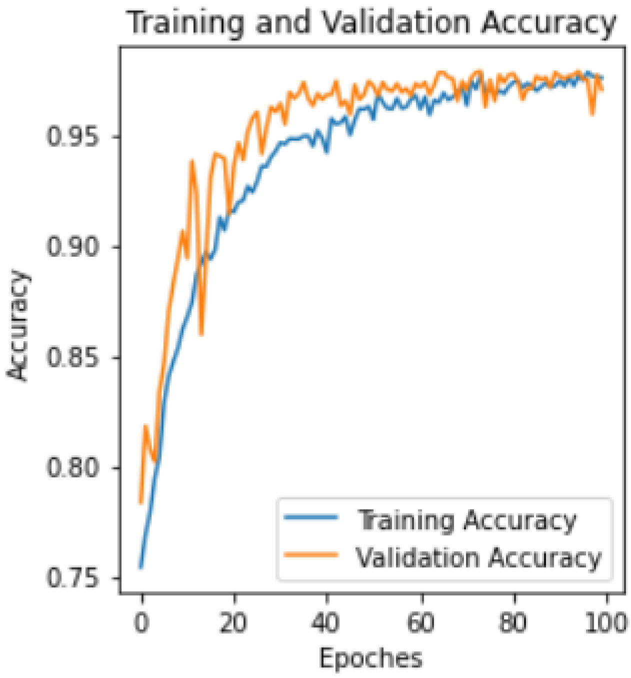


<Figure size 432x288 with 0 Axes>

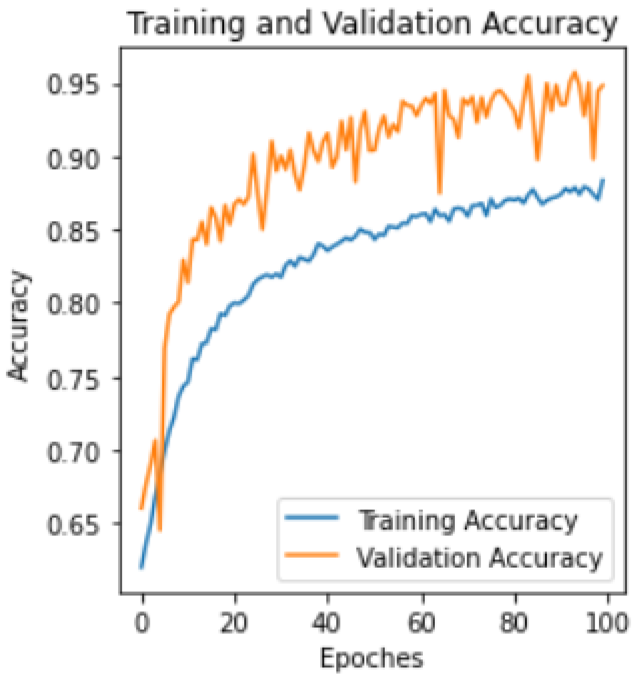
VGG16 ISIC 2019



OUR MODEL HAM10000



OUR MODEL ISIC 2019



CONCLUSION

In this paper, three pre-trained CNN models: MobileNet, ResNet-50, and VGG-16 based on transfer learning are compared using the HAM10000 dataset and ISIC 2019 dataset, taking advantage of ImageNet weights. Using Python programming language, we classified the 7 types of skin cancer images present in the HAM10000 dataset and 8 types of skin cancer images in the dataset. A different set of parameters were employed to evaluate the CNN model. Data augmentation was used to increase the number of images for better accuracy. The experimental results show promising classification accuracy with all the above-mentioned models. The results obtained by the three models demonstrate accuracies of 92.42%, 71.60%, and 87.70%, respectively for the HAM10000 dataset, and accuracies of 93.63%, 61%, and 75.24% respectively for the ISIC 2019 dataset. The best pre-trained model is tested on the ISIC 2019 dataset showing an accuracy of 93.63%.

We proposed a model and got an accuracy of 97.04% and 94.83% for the HAM10000 and ISIC 2019 datasets respectively. The proposed methodology using CNN represents a helpful tool to diagnose skin cancer disease accurately. Finally, the highest accuracy for our proposed model is tested on the HAM10000 dataset i.e. 97.04%.

FUTURE WORK

· **Different types of optimizers can be used e.g.** RMSprop, AdaDelta.

· Different datasets can be used for comparative study.

· Can use the CNN model with a cost-effective approach without using complicated systems and costly equipment.

· Own dataset can be created with better images.

BIBLIOGRAPHY

[1] Detection of melanoma in dermoscopic images by integrating features extracted using handcrafted and deep learning models.

Priti Bansal, Ritik Garg, Priyank Soni (2022)

[2] Multiclass Skin Lesion Classification Using a Novel Lightweight Deep Learning Framework for Smart Healthcare.

Long Hoang, Suk-Hwan Lee, Eung-Joo Lee, Ki-Ryong Kwon (2022)

[3] Optimized Convolutional Neural Network Models for Skin Lesion Classification.

Juan Pablo Villa-Pulgari , Anderson Alberto Ruales-Torres, Daniel Arias-Garzón, Mario Alejandro Bravo-Ortiz, Harold Brayan Arteaga-Arteaga, Alejandro Mora-Rubio, Jesus Alejandro Alzate-Grisales, Esteban Mercado-Ruiz , M. Hassaballah , Simon Orozco-Arias, Oscar Cardona-Morales and ReinelTabares-Soto (2021)

[4] BILSK: A bilinear convolutional neural network approach for skin lesion classification.

Camilo Calderon, Karen Sanchez, Sergio Castillo, Henry Arguello (2021)

[5] Multiclass skin cancer classification using EfficientNets – a first step towards preventing skin cancer.

Karar Ali, Zaffar Ahmed Shaikh, Abdullah Ayub Khan, Asif Ali Laghari (2021)

[6] Deep learning design for benign and malignant classification of skin lesions: a new approach.

Wessam M. Salamaa, Moustafa H. Aly (2021)

[7] A multi-class skin Cancer classification using deep convolutional neural networks.

Saket S. Chaturvedi, Jitendra V. Tembhurne, Tausif Diwan (2020)

[8] A Convolutional Neural Network Framework for Accurate Skin Cancer Detection.

Karl Thurnhofer-Hemsi, Enrique Domínguez (2020)

[9] Skin Lesion Analyser: An Efficient Seven-Way MultiClass Skin Cancer Classification Using MobileNet.

Saket S. Chaturvedi, Kajol Gupta, Prakash S. Prasad (2020)

[10] Transfer learning with class-weighted and focal loss function for automatic skin cancer classification.

Duyen N.T. Le, Hieu X. Le, Lua T. Ngo, Hoan T. Ngo (2020)

[11]. Multi-class skin lesion classification using prism- and segmentation-based fractal signatures.

José Ariel Camacho-Gutiérrez, Selene Solorza-Calderón, Josué Álvarez-Borrego b (2022)

[12]. CS-AF: A cost-sensitive multi-classifier active fusion framework for skin lesion classification.

Di Zhuang, Keyu Chen, J. Morris Chang (2022)

[13]. Hierarchy-aware contrastive learning with late fusion for skin lesion classification.

Benny Wei-Yun Hsua, Vincent S. Tseng (2022)

[14]. SSD-KD: A self-supervised Diverse Knowledge Distillation Method for Lightweight Skin Lesion Classification Using Dermoscopic Images.

Yongwei Wanga, Yuheng Wang, Tim K. Lee, ChunyanMiaog, Z. Jane Wang (2022)

[15]. Multi-features extraction based on deep learning for skin lesion classification.

SamiaBenyahia, BoudjelalMeftah, Olivier L´ezoray (2021)

[16]. Unsupervised Approaches for Out-Of-Distribution Dermoscopic Lesion Detection.

Max Torop, Sandesh Ghimire, Wenqian Liu, Dana H. Brooks, Octavia Camps, Milind Rajadhyaksha, Jennifer Dy, KivancKose (2021)

[17]. An approach for multiclass skin lesion classification based on ensemble learning.

Zillur Rahman, Md. Sabir Hossain, Md. Rabiul Islam, Md. Mynul Hasan, Rubaiyat Alim Hridhee. (2021)

[18]. Skin Lesions Classification into Eight Classes for ISIC 2019 Using Deep Convolutional Neural Network and Transfer Learning.

MOHAMED A. KASSEM, KHALID M. HOSNY, MOHAMED M. FOUAD (2020)

[19]. Skin cancer detection: Applying a deep learning-based model-driven architecture in the cloud for classifying dermal cell images.

Mohammad AliKadampur SulaimanAlRiyaee (2020)

[20].https://www.wcrf.org/cancer-trends/skin-cancer-statistics/#:~:text=skin%20cancer%20data-,Melanoma%20of%20skin%20is%20the%2017th%20most%20common%20cancer%20worldwide,the%20reporting%20of%20cancer%20statistics.

[21].https://www.scienceabc.com/eyeopeners/why-does-the-thickness-of-skin-vary-over-different-parts-of-the-body.html

[22]. https://www.who.int/news-room/fact-sheets/detail/cancer

[23]. https://www.ncbi.nlm.nih.gov/books/NBK441949/

[24].https://www.mayoclinic.org/diseases-conditions/skin-cancer/symptoms-causes/syc-20377605

[25]. Skin lesion classification using ensembles of multi-resolution EfficientNets with metadata

Nils Gesserta,b,∗ , Maximilian Nielsenb,c , Mohsin Shaikhb,c , René Werner b,c, Alexander Schlaefer (2020)

[26]. Improved skin lesion recognition by a Self-Supervised Curricular Deep Learning approach

Kirill Sirotkin, Marcos Escudero-Vinolo, Pablo Carballeira, Juan C. SanMiguel(2021)

[27]. https://www.skincancer.org/skin-cancer-information/basal-cell-carcinoma/#:~:text=Basal%20cell%20carcinoma%20(BCC)%20is,uncontrolled%20growth%20of%20basal%20cells.

[28]. WonDerM: Skin Lesion Classification with Fine-tuned Neural Networks

Yeong Chan Lee1 , Sang-Hyuk Jung1 , and Hong-Hee Won1(2019)

[29]. Skin Diseases Classification Using Deep Learning Methods

ANCA-LOREDANA UDRIȘTOIU,[1] ARIANA ELENA STANCA,[1] ALICE ELENA GHENEA,[2] CORINA MARIA VASILE,[2] MIHAELA POPESCU,[2] ȘTEFAN CRISTINEL

UDRIȘTOIU,[1] ANDREEA VALENTINA IACOB,[1] STEFAN CASTRAVETE,[4] LUCIAN GHEORGHE GRUIONU,[3] and GABRIEL GRUIONU[3]

[30]. Deep Learning-Based Automatic Assessment of Radiation Dermatitis in Patients with Nasopharyngeal Carcinoma

RuiyanNiBSc*TaZhouPhD*GeRenPhD*YuanpengZhangPhD*DongrongYangBSc*Victor C.W.TamPgDPH*Wan ShunLeungPhD*HongGeMD, PhD†SharaW.Y.LeePhD*JingCaiPhD

[31]. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[32]. Introduction to Convolutional Neural Networks (CNN)

Manav Mandal — May 1, 2021

[33]. Machine learning

From Wikipedia, the free encyclopedia

[34]. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition.

He K, Zhang X, Ren S, Sun J.

[35]. Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms

Morteza Heidari Seyedehnafiseh Mirniahari kandehei Abolfazl Zargari Khuzani Gopichandh Danala YuchenQiu Bin Zheng

[36]. Multi-features extraction based on deep learning for skin lesion classification

SamiaBenyahia, BoudjelalMeftah , Olivier L´ezoray (2022)

[37]. Analysis of the ISIC image datasets: Usage, benchmarks, and recommendations

Bill Cassidy, Connah Kendrick, Andrzej Brodzicki, Joanna Jaworek-Korjakowska, Moi Hoon Yap (2022)

[38]. Skin Lesion Classification Using Deep Neural Network

Guissous Alla Eddine

[39]. Skin Cancer Detection: A Review Using Deep Learning Techniques

Mehwish Dildar, Shumaila Akram, Muhammad Irfan, Hikmat Ullah Khan, Muhammad Ramzan, Abdur Rehman Mahmood, Soliman Ayed Alsaiari, Abdul Hakeem M Saeed, Mohammed Olaythah Alraddadi, and Mater Hussen Mahnashi

[40]. Deep learning design for benign and malignant classification of skin lesions: a new approach

Wessam M. Salamaa ,Moustafa H. Aly

[41]. M. E. Celebi et al**., A methodological approach to the classification of dermoscopy images, Comput. Med. Imaging Graph., 2007,**

DOI: 10.1016/j.compmedimag.2007.01.003.

[42]. Application of an artificial neural network in epiluminescence microscopy pattern analysis of pigmented skin lesions: a pilot study,

M. BINDER, A. STEINER, M. SCHWARZ, S. KNOLLMAYER, K. WOLFF, and H. PEHAMBERGER, Br. J. Dermatol., 1994, doi: 10.1111/j.1365-2133.1994.tb03378.x.

[43]. H. Dermatoscopy of pigmented Bowen's disease. J Am Acad Dermatol 62, 597–604 (2010).

Cameron, A., Rosendahl, C., Tschandl, P., Riedl, E. & Kittler

[44]. Dermatoscopy of facial actinic keratosis, intraepidermal carcinoma, and invasive squamous cell carcinoma: a progression model. J. Am. Acad. Dermatol. 66, 589–597 (2012).

Zalaudek, I. et al.

[45]. H. Dermatoscopy of flat pigmented facial lesions. J EurAcad Dermatol Venereol 29, 120–127 (2015).

Tschandl, P., Rosendahl, C. & Kittler

[46]. Studying regression of seborrheic keratosis in lichenoid keratosis with sequential dermoscopy imaging. Dermatology 220, 103–109 (2010).

Zaballos, P. et al.

[47]. Moscarella, E. et al. Lichenoid keratosis-like melanomas.

J Am Acad Dermatol 65, e85, Van de (2011).

[48]. Dermoscopy of pigmented seborrheic keratosis: a morphological study. Arch Dermatol 138, 1556–1560 (2002).

Braun, R. P. et al.

[49]. J. Dermoscopy of dermatofibromas: a prospective morphological study of 412 cases. Arch Dermatol 144, 75–83 (2008).

Zaballos, P., Puig, S., Llambrich, A. &Malvehy,

[50]. D. Dermatoscopy in routine practice - 'chaos and clues'. Aust Fam Physician 41, 482–487 (2012).

Rosendahl, C., Cameron, A., McColl, I. & Wilkinson,

[51]. Improvement of early recognition of lentigo maligna using dermatoscopy. J. Am. Acad. Dermatol. 42, 25–32 (2000).

Schiffner, R. et al.

[52]. Dermoscopy of solitary angiokeratomas: a morphological study. Arch Dermatol 143, 318–325 (2007).

Zaballos, P. et al.

[53]. Dermoscopy of pyogenic granuloma: a morphological study. Br J Dermatol 163, 1229–1237 (2010).

Zaballos, P. et al.

[54]. Bcn20000: Dermoscopic lesions in the wild, arXiv preprint arXiv:1908.02288, 2019.

M. Combalia et al.,

[55]. Tutorial: Getting Started with Machine Learning in Python

https://www.stxnext.com/blog/getting-started-machine-learning-python/

[56]. https://pythonbasics.org/split-train-test/

[57]. ML Practicum: Image Classification

https://developers.google.com/machine-learning/practica/image-classification#:~:text=Image%20classification%20is%20a%20supervised,the%20input%20to%20the%20model.

[58]. Image classification using Machine Learning made simple

https://towardsdatascience.com/image-classification-using-machine-learning-made-simple-cf7428a85bee

[59]. https://viso.ai/deep-learning/pattern-recognition/

[60]. THE RESEARCH OF THE FAST SVM CLASSIFIER METHOD

YUJUN YANG 1,2,3, JIANPING LI 1, YIMEI YANG 2,

[61]. SVM Machine Learning Tutorial – What is the Support Vector Machine Algorithm, Explained with Code Examples (2020)

https://www.freecodecamp.org/news/svm-machine-learning-tutorial-what-is-the-support-vector-machine-algorithm-explained-with-code-examples/

[62]. Skin lesions classification using deep learning Based on dilated convolution (2020)

Md. AminurRab Rahul, M. Hamed Mozaffari, Dr. Won Sastle Lee, Dr. EneaParimbelli

[63]. An Enhanced Technique of Skin Cancer Classification using Deep Convolution Neural Network with Transfer Learning Model. (2021)

Aryan Mobiny, Aditi Singh, and Hien Van Nguyen

[64]. Deep Neural Frameworks improve the accuracy of general practitioners in the classification of pigmented skin lesions (2020)

Maximilliano lucius, Jorge De Ali, Jose Antonio De Ali, Martin Belvisi, Luciana Radizza, Marisa Lanfranconi, Victoria Lorenzatti, Carlos M. Galmarini

[65]. Risk-Aware Machine Learning Classifier for Skin Lesion Diagnosis (2019)

Aryan Mobiny, Aditi Singh, and Hien Van Nguyen

[66]. A STUDY ON PYTHON PROGRAMMING LANGUAGE (2020)

Dr.D. Lakshminarayanan1 and S. Prabhakaran2