People’s Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
Echahid Hamma Lakhdhar University, El Oued

Department of Computer Science
Faculty of Exact Sciences

A Hybrid Deep Learning Machine Learning Approach for Damage Classification

Submitted by:
Gheraissa saad

Supervisor: Dr. Ben Ali Abdelkamel

Board of Examiners

Chairman:	Echahid Hamma Lakhdhar University, El Oued
Supervisor: **Dr. Ben Ali Abdelkamel**	Echahid Hamma Lakhdhar University, El Oued
Examiner:	Echahid Hamma Lakhdhar University, El Oued

Academic Year: 2021/2022

Thankful

We would like to thank from the bottom of our hearts, and above all, Almighty God for giving us the strength, the will, and the courage to do this work.

We would like to warmly and successively thank all those who have contributed from near and far to the realization of this final study project. We thank **Dr. Ben Ali Abdelkmel** for his confidence and his effective supervision and we express to him our deepest gratitude for sharing his knowledge and his kindness, as well as his working methods, especially for its scientific rigor.

We would like to thank the members of the jury for accepting to judge our works.

We also express our gratitude to all those who have cooperated in our training, especially the teachers of the University Information Technology Department Shahidat Hama Al-Akhdar Al-Wadi for our colleagues from semester 2021-2022. We also thank all those who participated directly or indirectly in the development of this work.

Abstract

A critical task after earthquakes is to quickly and precisely detect and identify damaged areas? Several soft computing techniques have been established to analyze remote sensing (RS) images for earthquake damage classification. The mono-temporal methods for damage classification, that use RS data collected after disasters, are mostly based on supervised learning approaches. The performance of these methods is largely depending on powerful learning feature representations. Machine learning (ML) techniques that use hand-crafted features can achieve satisfactory performance, but they do not generalize well. Deep learning (DL) techniques, in particular the convolutional neural networks (CNNs) have demonstrated its ability of deriving more powerful feature representations in many domains. Our main goal in this study is the investigation of a hybrid feature extraction approach for post-earthquake damage classification. Our hybrid approach combines features derived from a pretrained CNN model with the conventional hand-crafted features in order to improve the classification performance. We validated our proposal on two large datasets captured from different earthquake events and several geographic locations. The experimental validation showed that the performance gain apported by our hybrid approach is very significant.

Keywords: Remote sensing; earthquake damage classification; hybrid feature extraction; pretrained CNN; Gabor filter.

Résumé

Une tâche critique après les tremblements de terre est de détecter et d'identifier rapidement et précisément les zones endommagées ? Plusieurs techniques informatiques ont été établies pour analyser les images de télédétection (ou en anglais Remote Sensing [RS] images) pour la classification des dommages causés par les tremblements de terre. Les méthodes mono-temporelles pour la classification de dommages, qui utilisent des données collectées après les catastrophes, sont ordinairement basées sur des approches d'apprentissage supervisé. La performance de ces méthodes dépend largement de la puissante de représentations des caractéristiques d'apprentissage. Les techniques d'apprentissage automatique (en anglais Machine Learning ou ML) qui utilisent des caractéristiques de bas niveau (provenant d'experts) puissent atteindre des performances satisfaisantes, mais elles ne se généralisent pas toujours bien. Les techniques d'apprentissage en profondeur (en anglais Deep Learning ou DL), en particulier le réseau de neurones convolutifs (en anglais Convolutional Neural Networks ou CNNs), ont démontré leur grande capacité à dériver des représentations de caractéristiques plus puissantes dans de nombreux domaines. Notre objectif principal dans ce travail de mémoire de Master est l'étude d'une approche hybride d'extraction de caractéristiques pour la classification des dommages post-séisme. Notre approche hybride combine des caractéristiques dérivés d'un modèle CNN pré-entraîné et des caractéristiques de bas niveau calculées par des opérateurs spécifiques, en vue d'améliorer la performance de classification. Nous avons validé notre proposition sur deux grands ensembles de données capturés à partir de différents événements sismiques et de plusieurs emplacements géographiques. Les expériences de validation ont montré que le gain de performance apporté par notre approche hybride est très significatif.

Mots-clés : Télédétection ; classification des dommages causés par les tremblements de terre ; extraction hybride des caractéristiques ; CNN pré-entraîné ; Filtre de Gabor.

ملخص

تتمثل المهمة الحاسمة بعد الزلازل في الكشف عن المناطق المتضررة وتحديدها بسرعة وبدقة؟ تم إنشاء العديد من التقنيات الحسابية لتحليل صور الاستشعار عن بعد (RS) لتصنيف أضرار الزلزال. عادةً ما تعتمد الطرق أحادية الوقت لتصنيف الأضرار، والتي تستخدم البيانات التي تم جمعها بعد الكوارث، على مناهج التعلم الخاضع للإشراف. يعتمد أداء هذه الأساليب إلى حد كبير على التمثيلات القوية لخصائص التعلم. يمكن أن تحقق تقنيات التعلم الآلي (ML) التي تستخدم ميزات منخفضة المستوى (مصممة طرف خبراء) أداءً مرضيًا، لكنها لا تعمم دائمًا بشكل جيد. أثبتت تقنيات التعلم العميق(DL) ، ولا سيما الشبكة العصبية التلافيفية (CNN) ، قدرتها الكبيرة على اشتقاق تمثيلات أكثر قوة في العديد من المجالات. هدفنا الرئيسي هو دراسة نهج استخراج الميزات الهجينة لتصنيف أضرار ما بعد الزلزال. يجمع نهجنا الهجين بين الميزات المشتقة من نموذج CNN والميزات منخفضة المستوى المحسوبة بواسطة مشغلين محددين، من أجل تحسين أداء التصنيف. لقد تحققنا من صحة اقتراحنا باستعمال مجموعتي بيانات كبيرتين تم التقاطهما من أحداث زلزالية مختلفة ومواقع جغرافية متعددة. أظهرت تجارب التحقق من الصحة أن مكاسب الأداء التي يوفرها نهجنا المختلط مهمة للغاية.

الكلمات الرئيسية: الاستشعار عن بعد؛ تصنيف الضرر الناجم عن الزلازل؛ استخراج السمة الهجينةCNN ؛ مدربة مسبقًا؛ مرشح غابور.

Contents

Thankful I

Abstract II

Résumé III

ملخص IV

General introduction 2

Chapter Ⅰ **5**

Image classification and Feature extraction **5**

1 Introduction 6

2 What is Image Classification? 6

2-1 Unsupervised classification 6

2-2 Supervised classification 7

3 Image classification methods 7

3.1 How Does Image Classification Work? 8

4 Image Classification using Machine Learning 8

4-1 Recent Advances in Image Classification 8

4-2 Advantages of Deep Learning vs. traditional Image Processing 9

4-3 The Success of CNN Image Classification 10

5 What is Feature Extraction? 11

5-1 Applications of Feature Extraction 11

5-2 Methods for feature extractions 12

Method #1: Grayscale Pixel Values as Features 12

Method #2: Mean Pixel Value of Channels 12

Method #3: Extracting Edge Features 12

6 Filters for image feature extraction 14

7 Conclusion 15

Chapter Ⅱ **16**

Machine learning and Deep learning **16**

1 Introduction 17

2 Machine Learning 18

2-1 Support Vector Machine (SVM) 20

3 Deep Learning 21
3-1 CNN Architecture and Layers 22
3-2 Applications of deep learning 24
3-3 Deep learning methods 24
3-4 Deep learning examples 25
4 Differences between deep learning and machine learning 26
5 Transfer leaning 27
5-1 Models That Have Been Pre-Trained 28
5-2 VGG-16 28
6 Conclusion 31
Chapter Ⅲ 32
Damage Classification using hybrid Deep learning Machine learning 32
1 Introduction 33
2 Development environment and tools used 33
2-1 Used Materials 33
2-2 Google Colab 33
2-3 Python language 35
2-4 Dataset 36
3 A previous study 37
4 Our proposal: Combining CNN and Gabor-based feature representations 39
4-1 Deep learning (CNN Image classification using transfer learning, VGG-19) 40
4-1-1 Feature extraction algorithms 41
4-1-2 Pre-trained CNN: VGG-19 41
4-2 Machine learning (Gabor filter & SVM Algorithm) 43
4-3 Combining CNN and Gabor features 45
4-4 Comparison with the study presented by Settou et al. [12] 47
5 Conclusion 47
General conclusion 48
Appendix A 50
Features extraction using Gabor filter and CNN, then classification with SVM 50
References 51

General introduction

Natural disasters, such as earthquakes, volcanic eruptions, tsunamis, landslides, floods, storms, cyclones, thunderstorms, etc., are "unexpected" and very costly in terms of loss of human life and resources. A critical task after a natural disaster is to quickly and precisely detect and identify damaged areas? Several soft computing techniques have been developed to analyze remote sensing (RS) images in order to classify damage after disasters.

So-called mono-temporal techniques that only use RS data collected after disasters are fast and efficient. They are based on **supervised learning approaches** which consist of two main modules: the **feature extraction module** and the **classification module**. The performance of damage classification is highly dependent on the representations and feature extraction methods used. The two major supervised learning approaches: **Machine Learning** (**ML**) and **Deep Learning** (**DL**) have been applied for damage classification.

In ML, an algorithm analyzes a set of data (images) in order to build models that will make it possible to classify new data. ML algorithms involve two distinct but complementary steps, feature extraction and then classification. Feature extraction is performed by specific operators on the image to transform the original spectral feature space into abstract representations that can be easily separated by a classifier, such as SVM (Support Vector Machines), ANN (Artificial Neural Network), DT (for Decision Trees), KNN (for K-Nearest-Neighbors), etc. Figure 1 illustrates the feature extraction step in ML. The performance of the classifier employed is strongly influenced by the transformations applied and the spatial features extracted. Although ML algorithms are efficient, feature extraction methods are difficult to design and require domain expertise (hand-crafted features).

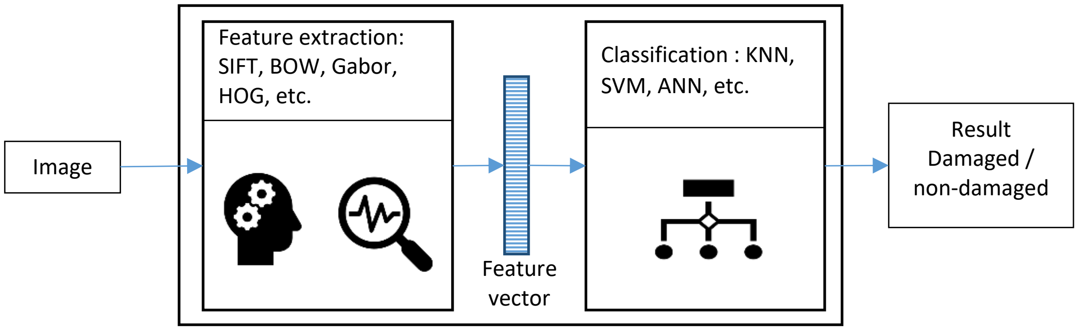

Figure 1: Feature extraction in ML.

In DL algorithms, a subset of ML, feature extraction is performed automatically by the algorithm itself. Figure 2 illustrates feature extraction in DL. DL is based on artificial neural networks (NNs), made up of thousands of units, "neurons", which each perform small, simple operations. The results of a first layer of "neurons" serve as input for the calculation of a second layer and so on. DL methods automatically learn hierarchical representations in deep architectures for classification. The goal is to discover more abstract features in the higher levels of the representation, using NNs that easily separate the various explanatory factors from the data. Recently, the convolutional neural network (CNN) has demonstrated its ability of deriving powerful feature representations in many domains.

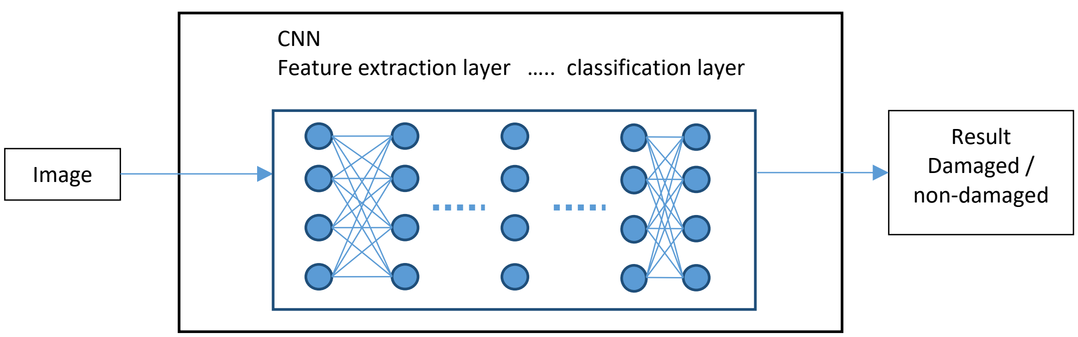

Figure 2: Feature extraction in DL.

Goals of our study

The objective of this Master's memory project is to hybridize DL techniques with ML techniques for damage classification, in order to improve performance. To do so:

- First, we prepare a feature vector derived from the higher layers of a pretrained CNN model, such as AlaxNet, GoogleNet, VGG16/19, ResNet, etc. This step goes through the two phases of DL, train and test.
- In a second step, we prepare hand-crafted (or ML) features calculated by image analysis techniques such as Gabor filters.
- In the third step, we combine the two CNN and ML feature vectors to build the input data of a supervised classifier such as SVM.

In this work, we investigate this idea for earthquake damage classification. Two large datasets are employed for validation.

Organization

This master memory is organized as follows:

– The first chapter presents the basic concepts and notions in image analysis and then the methods for feature extraction and classification.

– The second chapter is devoted to ML and DL. We will explain the relationship between ML and DL, and then introduce the applications and different approaches of DL. In addition, we will expose the CNN network architecture and several different models.

– The third chapter is devoted to the presentation of our general hybridization approach of ML and DL features and to the experimental validation of this approach for earthquake damage classification.

– Finally, we end with a general conclusion giving possible improvements as well as future directions.

Chapter Ⅰ

Image classification and Feature extraction

1 Introduction

Image classification is defined as the task of classifying an image from a fixed set of categories. It is a very active research topic in today's society and a most important direction in the field of image processing research. Many other computer vision challenges such as object detection and segmentation can be reduced to image classification.

In this chapter, we will explain relevant ML algorithms for image classification which operate in two complementary phases, feature extraction and classification. We will present different methods and well-known techniques in the literature for each phase.

Support Vector Machine (SVM) is a very powerful classification model in ML. Convolutional Neural Networks (CNN) is a type of feedforward neural network that includes convolution calculation and has a deep structure. It is one of the representative algorithms of DL. Taking SVM and CNN as examples, this chapter compares and analyzes the traditional machine learning and deep learning image classification algorithms. Today, the use of CNNs is the state-of-the-art method for image classification.

2 What is Image Classification?

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.

Image classification techniques are mainly divided into two categories: Supervised and unsupervised techniques [1].

2-1 Unsupervised classification

An unsupervised classification technique is a fully automated method that does not leverage training data. This means ML algorithms are used to analyze and cluster unlabeled datasets by discovering hidden patterns or data groups without the need for human intervention [1].

With the help of a suitable algorithm, the particular characterizations of an image are recognized systematically during the image processing stage. Pattern recognition and image clustering are two of the most common image classification methods. Two popular algorithms used for unsupervised image classification are ‘K-means’ and ‘ISODATA.’

- **K-means** groups objects into *k* groups based on their characteristics. It is also called “clusterization.” K-means clustering is one of the simplest and very popular unsupervised ML algorithms [1].
- **ISODATA** (for “Iterative Self-Organizing Data Analysis Technique”) includes iterative methods that use Euclidean distance as the similarity measure to cluster data elements into different classes. While K-means assumes that the number of clusters is known a priori (in advance), ISODATA allows for a different number of clusters [1].

2-2 Supervised classification

Supervised image classification methods use previously classified reference samples (the ground truth) in order to train the classifier and subsequently classify new, unknown data. Therefore, the supervised image classification is the task of visually choosing samples of training data within the image and allocating them to pre-chosen categories, including vegetation, roads, water resources, and buildings. This is done to create statistical measures to be applied to the overall image [1].

3 Image classification methods

Two of the most common methods to classify the overall image through training data are ‘maximum likelihood’ and ‘minimum distance.’ For instance, ‘maximum likelihood’ classification uses the statistical traits of the data where the standard deviation and mean values of each textural and spectral indices of the picture are analyzed first. Later, the likelihood of each pixel to separate classes is calculated by means of a normal distribution for the pixels in each class. Moreover, a few classical statistics and probabilistic relationships are also used. Eventually, the pixels are marked to a class of features that show the highest likelihood [1].

3.1 How Does Image Classification Work?

A computer analyzes an image in the form of pixels. It does it by considering the image as an array of matrices with the size of the matrix reliant on the image resolution. Put simply, image classification in a computer's view is the analysis of this statistical data using algorithms. In digital image processing, image classification is done by automatically grouping pixels into specified categories, so-called "classes".

The algorithms segregate the image into a series of its most prominent features, lowering the workload on the final classifier. These characteristics give the classifier an idea of what the image represents and what class it might be considered into. The characteristic or feature extraction process is the most important step in categorizing an image as the rest of the steps depend on it.

Image classification, particularly supervised classification, is also reliant hugely on the data fed to the algorithm. A well-optimized classification dataset works great in comparison to a bad dataset with data imbalance based on class and poor quality of images and annotations [1].

4 Image Classification using Machine Learning

Image recognition with ML leverages the potential of algorithms to learn hidden knowledge from a dataset of organized and unorganized samples (Supervised Learning). The most popular ML technique is deep learning (DL), where a lot of hidden layers are used in a model [1].

4-1 Recent Advances in Image Classification

With the advent of DL, in combination with robust AI hardware and GPUs, outstanding performance can be achieved on image classification tasks. Hence, DL brought great successes in the entire field of image recognition, face recognition, and image classification algorithms achieve above human-level performance and real-time object detection.

Additionally, there is been a huge jump in algorithm inference performance over the last few years. For example, in 2017, the Mask R-CNN algorithm was the fastest real-time object detector on the MS COCO benchmark, with an inference time of 330 ms per frame. In comparison, the YOLOR algorithm, which was released in 2021, achieves inference times of 12 ms on the same benchmark, thereby overtaking the infamous YOLOv4 and YOLOv3 DL algorithms [1].

4-2 Advantages of Deep Learning vs. traditional Image Processing

In comparison to the conventional computer vision approach in early image processing around two decades ago, DL requires only the knowledge of engineering of a ML tool. It does not need expertise in particular machine vision areas to create handcrafted features.

In any case, DL requires manual data labeling to interpret good and bad samples, which is known as image annotation. The process of gaining knowledge or extracting insights from data labeled by humans is called supervised learning. And the process of creating such labeled data to train AI models needs tedious human work — for instance, to annotate regular traffic situations in autonomous driving. However, nowadays, we have large datasets with millions of high-resolution labeled data of thousands of categories such as ImageNet, LabelMe, Google OID, or MS COCO [1].

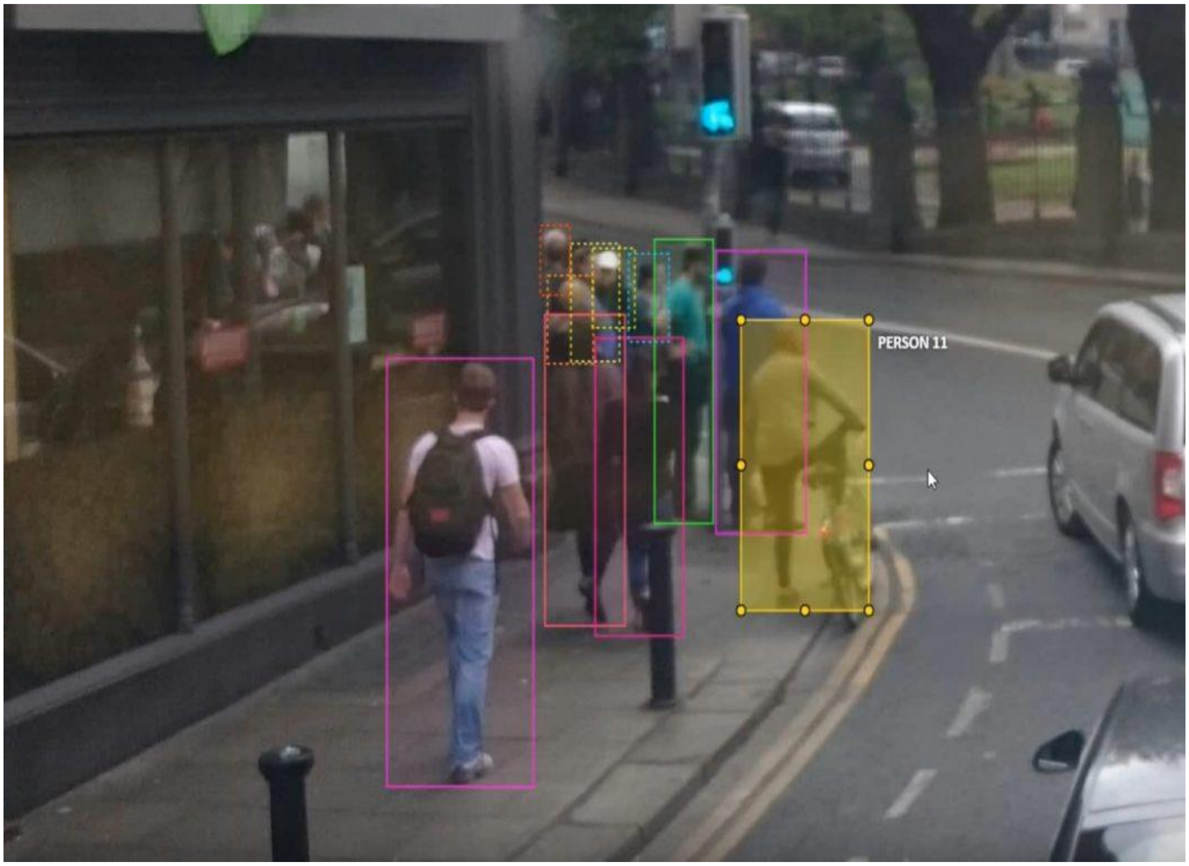

Figure 1.1: Example of manual image annotation for supervised training of DL algorithms in a video frame [1].

4-3 The Success of CNN Image Classification

The task of categorizing images into one or more predefined classes is known as image classification. Although humans have an instinctive and habitual ability to categorize images, it is much more challenging for an automated system to recognize and classify images, it is much more challenging for an automated system to recognize and classify images, and from Among deep neural networks (DNN), the convolutional neural network (CNN) has demonstrated excellent results in computer vision tasks, especially in image classification.

Convolutional Neural Network (CNN, or ConvNet) is a special type of multi-layer neural network inspired by the mechanism of the optical and neural systems of humans

In 2012, a large deep convolutional neural network called AlexNet showed excellent performance on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), this marked the start of the broad use and development of convolutional neural network models (CNN) such as VGGNet, GoolgeNet, and many more.

let’s Now talk about feature extraction.

5 What is Feature Extraction?

Feature extraction is a part of the dimensionality reduction process, in which, an initial set of the raw data is divided and reduced to more manageable groups. So, when you want to process it will be easier. The most important characteristic of these large data sets is that they have a large number of variables. These variables require a lot of computing resources to process. So, Feature extraction helps to get the best feature from those big data sets by selecting and combining variables into features, thus, effectively reducing the amount of data. These features are easy to process, but still able to describe the actual data set with accuracy and originality [2].

5-1 Applications of Feature Extraction

- **Bag of Words:** Bag-of-Words is the most used technique for natural language processing. In this process they extract the words or the features from a sentence, document, website, etc. and then they classify them into the frequency of use. So, in this whole process feature extraction is one of the most important parts [2].

- **Image Processing:** Image processing is one of the best and most interesting domains. In this domain basically you will start playing with your images in order to understand them. So here we use many techniques which includes feature extraction as well and algorithms to detect features such as shaped, edges, or motion in a digital image or video to process them [2].

- **Auto-encoders:** The main purpose of the auto-encoders is efficient data coding which is unsupervised in nature. this process comes under unsupervised learning.

So, Feature extraction procedure is applicable here to identify the key features from the data to code by learning from the coding of the original data set to derive new ones [2].

5-2 Methods for feature extractions

Method #1: Grayscale Pixel Values as Features

The simplest way to create features from an image is to use these raw pixel values as separate features.

Consider the same example for our image above (the number ‘8’) – the dimension of the image is 28 x 28.

Can you guess the number of features for this image? The number of features will be the same as the number of pixels! Hence, that number will be 784.

Now here’s another curious question – how do we arrange these 784 pixels as features? Well, we can simply append every pixel value one after the other to generate a feature vector [3].

Method #2: Mean Pixel Value of Channels

While reading the image in the previous section, we had set the parameter *‘as_gray = True’*. So, we only had one channel in the image and we could easily append the pixel values. Let us remove the parameter and load the image again:

Method #3: Extracting Edge Features

Consider that we are given the below image and we need to identify the objects present in it:

Figure 1.2: Image for identify the objects [3].

You must have recognized the objects in an instant – a dog, a car and a cat. What are the features that you considered while differentiating each of these images? The shape could be one important factor, followed by color, or size. What if the machine could also identify the shape as we do?

A similar idea is to extract edges as features and use that as the input for the model. I want you to think about this for a moment – how can we identify edges in an image? Edge is basically where there is a sharp change in color. Look at the below image [3].

Figure 1.3: Image for identify the Edges [3].

I have highlighted two edges here. We could identify the edge because there was a change in color from white to brown (in the right image) and brown to black (in the left). And as we know, an image is represented in the form of numbers. So, we will look for pixels around which there is a drastic change in the pixel values [3].

6 Filters for image feature extraction

Before we go to image filtering, let's first explain what is the filter in general.

6-1 Filter

The Filter is a device or process that removes some unwanted components or features from a signal and is capable of passing or amplifying certain frequencies while attenuating other frequencies or frequency ranges, however filters do not work only in the frequency domain especially in the field of image processing where there are many other purposes for filtering, that's everything about the filter in general context [15].

Now we will talk a little about image filtering.

6-2 Image filtering

Image filtering is a modify pixels in image based on some functions of a local neighborhood of the pixels. There are many filters, including the following

- **Gabor filter**

The Gabor filter texture analysis method is an important transformation-based metamaterial feature extraction method, and by selecting a specific Gabor function and designing the Gabor filter, omnidirectional and omnidirectional scale feature extraction of the image is achieved. In a two-dimensional space, using a trigonometric function (such as the sine function) and a Gaussian function [14].

- **Statistical filter**

  Image segmentation, object identification, and bandwidth compression all require the use of edge extraction techniques as a preprocessing step in the extraction of image features. Because of the natural spatial texture of the scene background, traditional edge extractors such as Sobel and Laplacian filters produce images with a high degree of clutter. To address this issue, a statistical filter was created that boosts local grey level activity around objects while reducing background contributions. In a neighborhood modification process, the statistical filter is used to replace the central pixel with the third central moment computed from the surrounding neighborhood [16].

- **Gaussian Filter**

Gaussian filtering is used to blur images and remove noise and detail. In one dimension, the advantage of a Gaussian filter is that its Fourier transform is also a Gaussian distribution centered on zero frequency (with positive and negative frequencies at both sides). The width of the filter can then be adjusted to control the effectiveness of the low-pass nature of the filter. Also, with a Gaussian filter, the attenuation of higher frequency components, and thus their relative removal, is more effective than with moving-average filters. The inherent statistical nature of fluctuations in many acquired measurement distributions may also be reflected by the Gaussian filter [16].

7 Conclusion

In this chapter, we cover everything you need to know about image classification – the computer vision task of identifying what an image represents. Today, the use of convolutional neural networks (CNN) is the state-of-the-art method for image classification

Chapter Ⅱ

Machine learning and Deep learning

1 Introduction

Machine learning, a sub-class of artificial intelligence, is self-learning based on algorithms that mean the system learns from its experience. For instance, the type of data given input to the system learns the pattern and responds from its learning at the output. In this case, the system becomes smart, smarter, and smartest with time without human involvement. It uses a statistical learning algorithm that automatically learns and improves without human help.

On the other side in a deep learning system, it learns from its experience but a large database or large information provided at input**.** Deep is the term that refers to several layers in between the input and output of a neural network whereas in shallow neural networks maximum of two layers are present in between the input and output neural network.

While there are many differences between these two subsets of artificial intelligence, But the main deferent is Machine learning requires more ongoing human intervention to get results. Still, Deep learning is more complex to set up but requires minimal intervention thereafter.

In this chapter, we will delve a lot into machine learning and deep learning, and we will be exposed to the most important modern methods in artificial intelligence and their efficiency in classification.

2 Machine Learning

Machine learning is a field of study which allows machines to learn from data or experience and make a prediction based on the experience. It enables the computers or the machines to make data-driven decisions rather than being explicitly programmed for carrying out a certain task. These programs or algorithms are designed in a way that they learn and improve over time when are exposed to new data [4].

➢ **Types of machine learning**

ML can be broadly divided into 3 subcategories [4], as illustrated in Figure 2.1.

1. Supervised learning

In supervised learning, we have a labeled data containing an input X and a label Y. In supervised learning, the task is to find the mapping between the input variable (X) called the independent variable and output variable (Y) called the dependent variable. Supervised learning can further be divided into two types of tasks.

- Regression: Regression problem is when the output variable is continuous and real value. For example, price, weight, etc.
- Classification: Classification is a problem when the output variable is a category, such as “red” or “blue” or “disease” or “no disease”.

2. Unsupervised learning

Unsupervised learning is where you only have input data (X) and no corresponding output variables. The goal of unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data. Unsupervised learning can further be divided into two types of tasks:

- Clustering: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.

- Association: An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

3. Reinforcement learning

Reinforcement learning directly takes inspiration from how human beings learn from data in their lives. It features an algorithm that improves upon itself and learns from new situations using a trial-and-error method. Favorable outputs are encouraged or 'reinforced', and non-favorable outputs are discouraged or 'punished' [19].

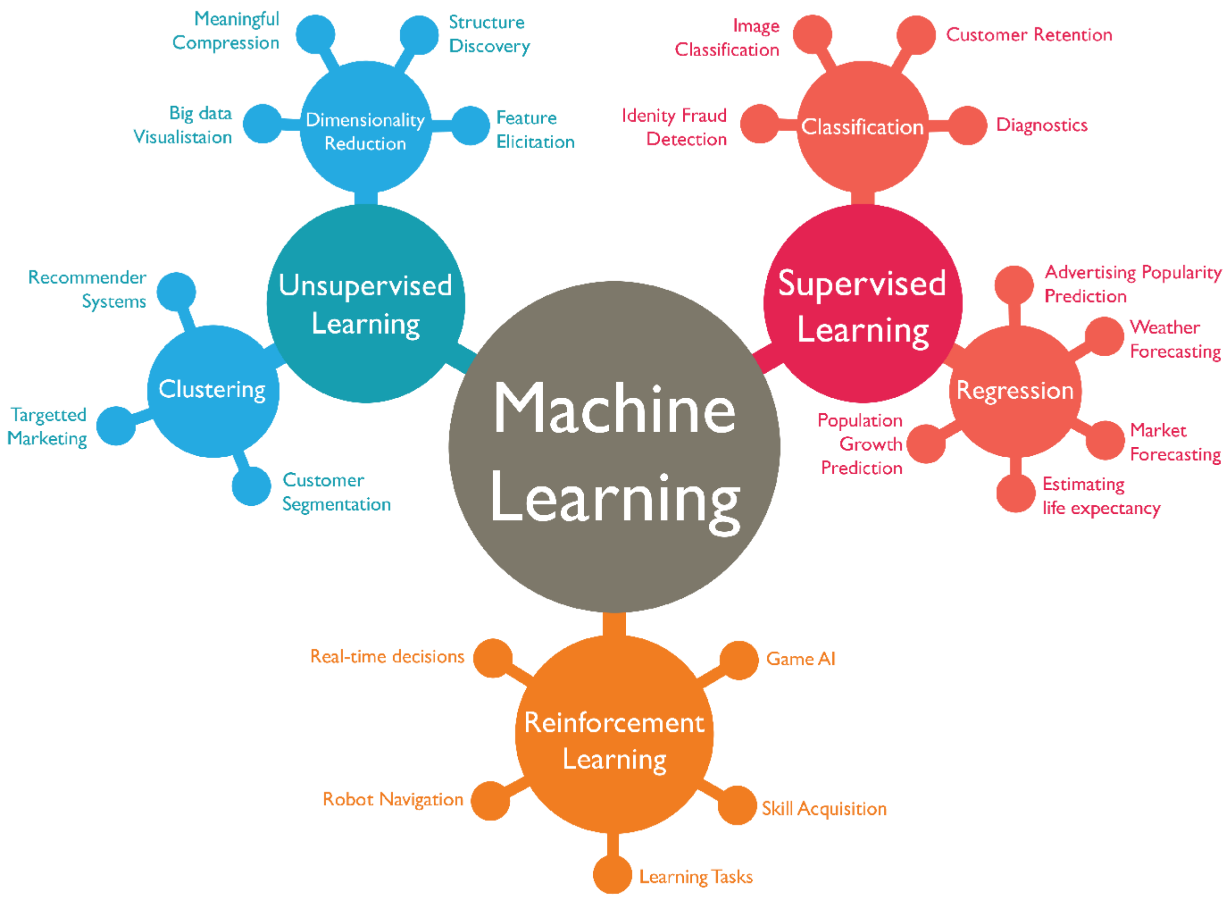

Figure 2.1: Machine learning categories with few applications [4].

Classification issues are considered one of the most important issues in the world of machine learning because of their importance in real life in several areas Researchers invented many algorithms that contributed to addressing many problems, including the support vector machine algorithm, which is symbolized by SVM

2-1 Support Vector Machine (SVM)

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane as shown in **Figure 2.2** [20].

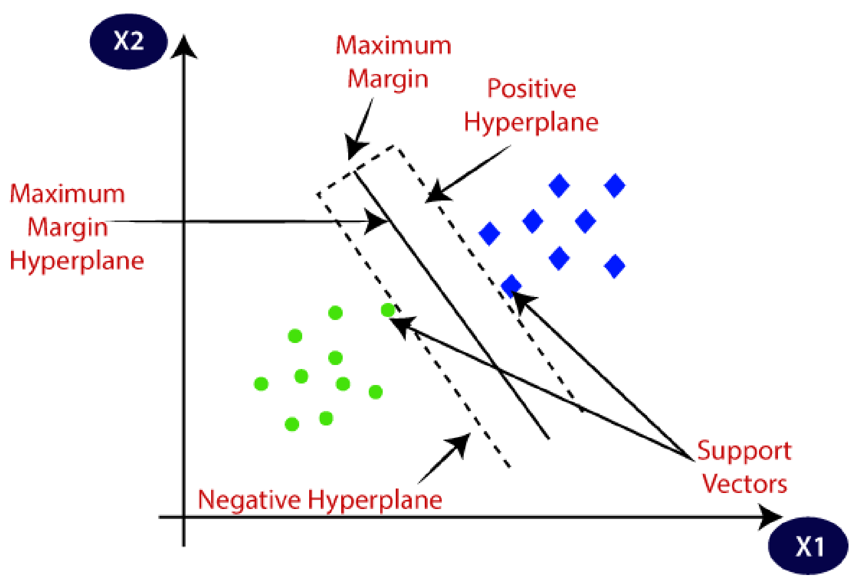

Figure 2.2: Linearly Separable Data points [20].

Now we will directly jump to deep learning without discussing machine learning algorithms.

3 Deep Learning

Deep Learning (DL) is a subfield of ML concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. DL algorithms can be regarded both as a sophisticated and mathematically complex evolution of ML algorithms. The field has been getting lots of attention lately and for good reason: Recent developments have led to results that were not thought to be possible before.

DL describes algorithms that analyze data with a logic structure similar to how a human would draw conclusions. Note that this can happen both through supervised and unsupervised learning. To achieve this, DL applications use a layered structure of algorithms called an artificial neural network (ANN). The picture **Figure 2.3** shows Simple artificial neural network (ANN) ,The design of such an ANN is inspired by the biological neural network of the human brain, leading to a process of learning that's far more capable than that of standard machine learning models [5].

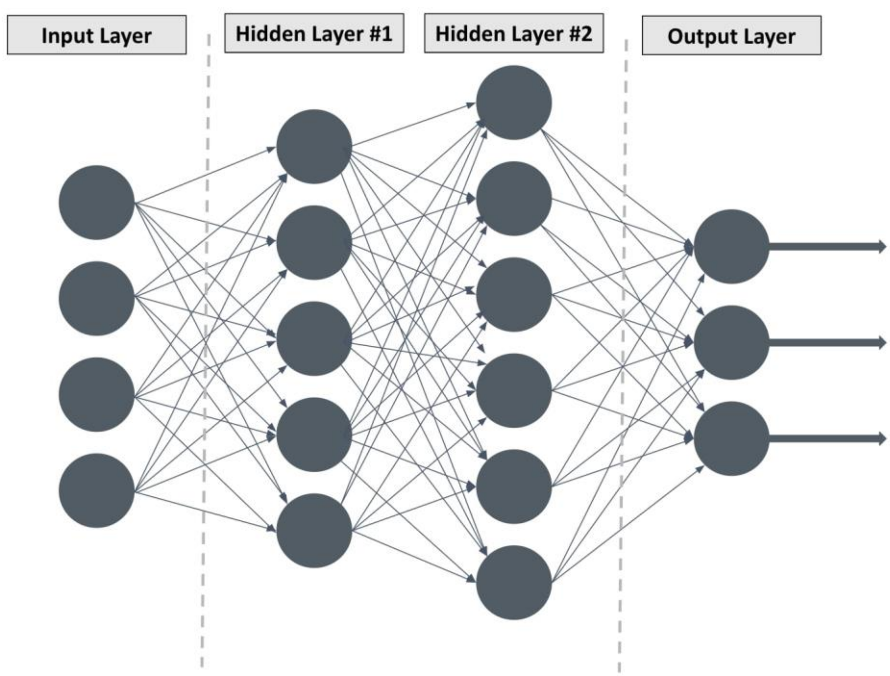

Figure 2.3: Simple artificial neural network [6].

3-1 CNN Architecture and Layers

A CNN is a machine learning framework that was created using machine learning concepts. Without the need for human intervention, CNNs can learn and train from data on their own.

In fact, there is only some pre-processing needed when using CNNs. They develop and adapt their own image filters, which have to be carefully coded for most algorithms and models. CNN frameworks have a set of layers that perform particular functions to enable the CNN to perform these functions.

The **Figure 2.4** shows the Concept of a neural network with the input values. The concept of neurons is based on human neurons. These are statistical functions that calculate the weighted average of inputs and apply an activation function to the result generated. Layers are a cluster of neurons, with each layer having a particular function [1].

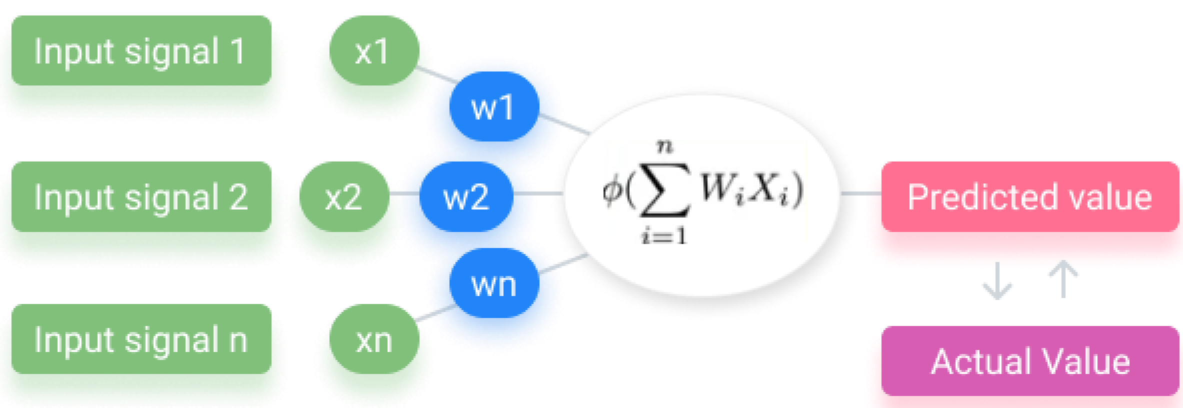

Figure 2.4: Concept of a neural network with the input values (green) and weights (blue) [1].

A CNN system can have anywhere from three to 150 layers, or even more: The term "deep neural networks" refers to the number of layers in the network. The output of one layer becomes the input of another. Vgg16 (16 layers) and VGG19 (19 layers) are examples of deep multi-layer neural networks like in **Figure 2.5** that clarify Concept of a Convolutional Neural Network (CNN) .

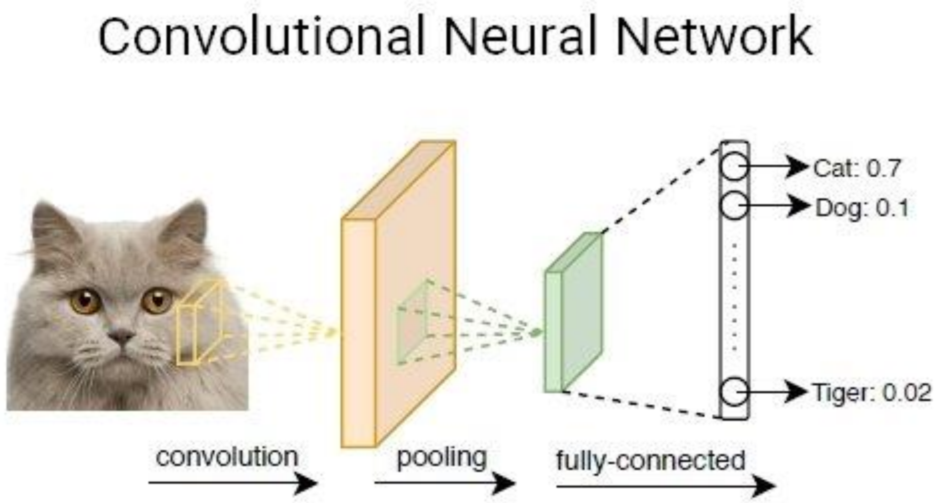

Figure 2.5: Concept of a Convolutional Neural Network (CNN) [1].

CNN layers can be of four main types: Convolution Layer, ReLu Layer, Pooling Layer, and Fully-Connected Layer [1].

- **Convolution Layer:** A convolution is the simple application of a filter to an input that results in an activation. The convolution layer has a set of trainable filters that have a small receptive range but can be used to the full-dept of data provided. Convolution layers are the major building blocks used in convolutional neural networks.
- **ReLu Layer:** ReLu layers, also known as Rectified linear unit layers, are activation functions applied to lower overfitting and build the accuracy and effectiveness of the CNN. Models that have these layers are easier to train and produce more accurate results.
- **Pooling Layer:** This layer collects the result of all neurons in the layer preceding it and processes this data. The primary task of a pooling layer is to lower the number of factors being considered and give streamlined output.
- **Fully-Connected Layer:** This layer is the final output layer for CNN models that flattens the input received from layers before it and gives the result.

3-2 Applications of deep learning

Applications of DL are vast and many of great technologies now use DL to improve the task. Some of the examples are [16]:

- Self-driving cars;
- Voice search and virtual assistants;
- Machine translation;
- Image caption generation;
- Colorization of Black and White Images;
- Game playing ai (Open Ai dota bot, google brain alpha go);
- Real-time object recognition in the image (Google lens).

3-3 Deep learning methods

Various methods can be used to create strong DL models. These techniques include learning rate decay, transfer learning, training from scratch and dropout [5].

- **Learning rate decay:** The learning rate is a hyperparameter -- a factor that defines the system or set conditions for its operation prior to the learning process -- those controls how much change the model experiences in response to the estimated error every time the model weights are altered. Learning rates that are too high may result in unstable training processes or the learning of a suboptimal set of weights. Learning rates that are too small may produce a lengthy training process that has the potential to get stuck. The learning rate decay method -- also called *learning rate annealing* or *adaptive learning rates* -- is the process of adapting the learning rate to increase performance and reduce training time. The easiest and most common adaptations of learning rate during training include techniques to reduce the learning rate over time.

- **Transfer learning**: This process involves perfecting a previously trained model; it requires an interface to the internals of a preexisting network. First, users feed the existing network new data containing previously unknown classifications. Once adjustments are made to the network, new tasks can be performed with more specific categorizing abilities. This method has the advantage of requiring much less data than others, thus reducing computation time to minutes or hours.

- **Training from scratch.** This method requires a developer to collect a large labeled data set and configure a network architecture that can learn the features and model. This technique is especially useful for new applications, as well as applications with a large number of output categories. However, overall, it is a less common approach, as it requires inordinate amounts of data, causing training to take days or weeks.

- **Dropout.** This method attempts to solve the problem of overfitting in networks with large amounts of parameters by randomly dropping units and their connections from the neural network during training. It has been proven that the dropout method can improve the performance of neural networks on supervised learning tasks in areas such as speech recognition, document classification and computational biology.

3-4 Deep learning examples

Because DL models process information in ways similar to the human brain, they can be applied to many tasks people do. DL is currently used in most common image recognition tools, natural language processing (NLP) and speech recognition software. These tools are starting to appear in applications as diverse as self-driving cars and language translation services.

Use cases today for DL include all types of big data analytics applications, especially those focused on NLP, language translation, medical diagnosis, stock market trading signals, network security and image recognition.

Specific fields in which DL is currently being used include the following [5]:

- **Customer experience (CX).** Deep learning models are already being used for chatbots. And, as it continues to mature, deep learning is expected to be implemented in various businesses to improve CX and increase customer satisfaction.

- **Text generation.** Machines are being taught the grammar and style of a piece of text and are then using this model to automatically create a completely new text matching the proper spelling, grammar and style of the original text.
- **Aerospace and military.** Deep learning is being used to detect objects from satellites that identify areas of interest, as well as safe or unsafe zones for troops.
- **Industrial automation.** Deep learning is improving worker safety in environments like factories and warehouses by providing services that automatically detect when a worker or object is getting too close to a machine.
- **Adding color.** Color can be added to black-and-white photos and videos using deep learning models. In the past, this was an extremely time-consuming, manual process.
- **Medical research.** Cancer researchers have started implementing deep learning into their practice as a way to automatically detect cancer cells.
- **Computer vision.** Deep learning has greatly enhanced computer vision, providing computers with extreme accuracy for object detection and image classification, restoration and segmentation.

4 Differences between deep learning and machine learning

ML means computers learning from data using algorithms to perform a task without being explicitly programmed. DL uses a complex structure of algorithms modeled on the human brain. This enables the processing of unstructured data such as documents, images and text [6].

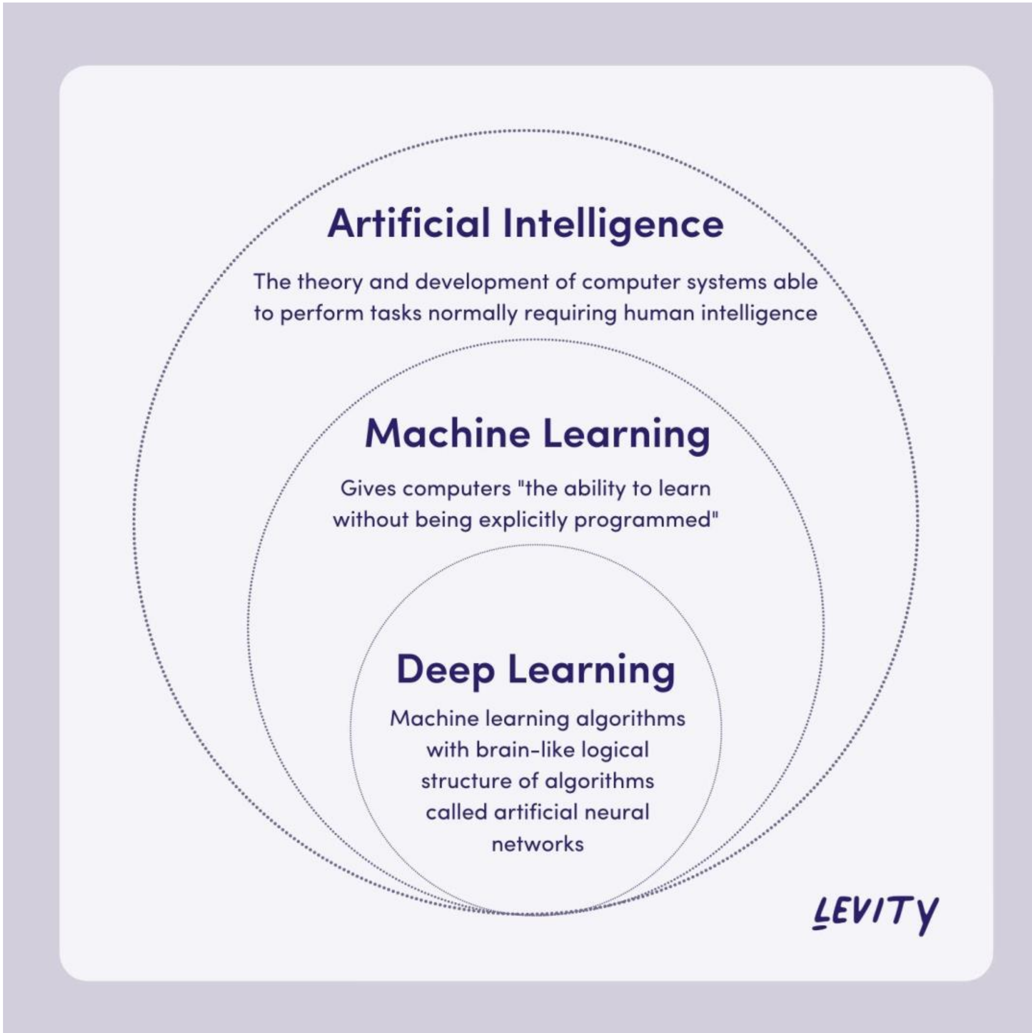

Figure 2.6: Machine Learning is a type of Artificial Intelligence. Deep Learning is an especially complex part of Machine Learning [6].

To break it down in a single sentence: DL is a specialized subset of machine learning which, in turn, is a subset of artificial intelligence. In other words, DL *is* machine learning.

5 Transfer leaning

The reuse of a pre-trained model on a new problem is known as transfer learning in machine learning. A machine uses the knowledge learned from a prior assignment to increase prediction about a new task in transfer learning. You could, for example, use the information gained during training to distinguish beverages when training a classifier to predict whether an image contains cuisine.

The reuse of a previously learned model on a new problem is known as transfer learning. It's particularly popular in deep learning right now since it can train deep neural networks with a small amount of data. This is particularly valuable in the field of data science, as most real-world situations do not require millions of labelled data points to train complicated models.

The knowledge of an already trained machine learning model is transferred to a different but closely linked problem throughout transfer learning. For example, if you trained a simple classifier to predict whether an image contains a backpack, you could use the model's training knowledge to identify other objects such as sunglasses[7].

5-1 Models That Have Been Pre-Trained

There are a number of popular pre-trained machine learning models available. The Inception-v3 model, which was developed for the ImageNet "Large Visual Recognition Challenge," is one of them." Participants in this challenge had to categorize pictures into 1,000 subcategories such as "zebra," "Dalmatian," and "dishwasher." Now at this point, you may be wondering what kind of pre-trained models you can use for your tasks. There are actually many models out there that allow you to plug-and-play, but in essence, the popular ones are the ones that may help you solve your relevant problems [7]:

- VGG (VGG-16 or VGG-19)
- GoogLeNet or InceptionV3
- Residual Network (ResNet50)

5-2 VGG-16

VGG-16 is a convolutional neural network that is 16 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database.

The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224. For more pretrained networks in MATLAB®, see Pretrained Deep Neural Networks.

You can use classify to classify new images using the VGG-16 network. Follow the steps of Classify Image Using GoogLeNet and replace GoogLeNet with VGG-16.

To retrain the network on a new classification task, follow the steps of Train Deep Learning Network to Classify New Images and load VGG-16 instead of GoogLeNet [8].

5-2-1 VGG-16 Architecture

Of all the configurations, VGG16 was identified to be the best performing model on the ImageNet dataset. Let's review the actual architecture of this configuration.

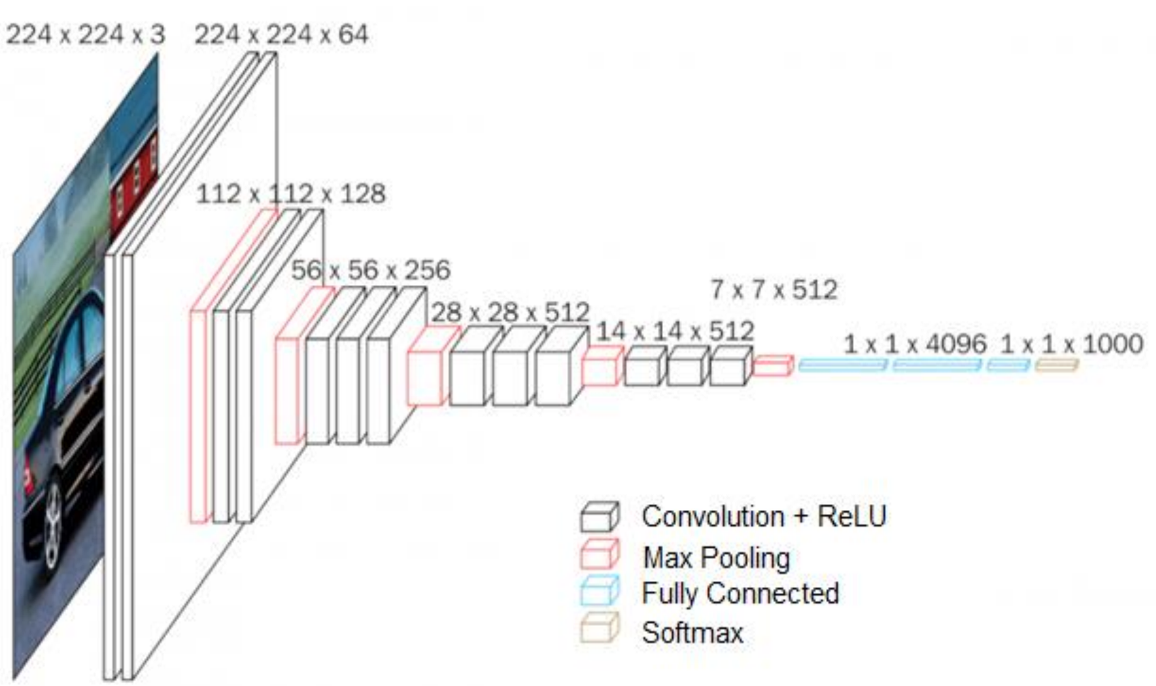

Figure 2.7: VGG-16 Architecture [8].

The input to cov1 layer is of fixed size 224 x 224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3 (which is the smallest size to capture the notion of left/right, up/down, center). In one of the configurations, it also utilizes 1×1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. layer input is such that the spatial resolution is preserved after convolution, i.e., the padding is 1-pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2.

Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has a different depth in different architectures): the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class).

The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks.

All hidden layers are equipped with the rectification (ReLU) non-linearity. It is also noted that none of the networks (except for one) contain Local Response Normalization (LRN), such normalization does not improve the performance on the ILSVRC dataset, but leads to increased memory consumption and computation time [8].

6 Conclusion

In this chapter, we briefly introduced machine learning with its types, and its different algorithms. Then, I explained briefly the principle of deep learning and its different models, then I gave some examples of that, all of that for the sake given the difference between them.

Chapter Ⅲ

Damage Classification using hybrid Deep learning Machine learning

1 Introduction

In this chapter, we describe and validate our proposal of combining CNN feature (DL approach) with hand-crafted features (ML approach) to ameliorate the performance of post-earthquake damage classification.

In the part of ML, we will be choosing the Support Vector Machines (SVM) algorithm using Gabor features. In the part of DL, we will choose the most famous and most accurate CNNs.

Our methods employed are validated on Aerial Dataset and UAV dataset for earthquake classification.

First let us take a look at Development environment and tools used.

2 Development environment and tools used

2-1 Used Materials

Table 3.1: Device characteristics.

Computer name	Processor	Operating system	Memory (RAM)
DELL E 6330	Intel(R) Core(TM) i5-3320M CPU @ 2.60GHz 2.60 GHz	64-bit operating system, x64-based processor	8.00 GB (7.88 GB usable)

2-2 Google Colab

Google Colab is becoming more and more popular for doing deep learning from home, preparing and sharing data science cases and working collaboratively with each other. Personally, I have been using Google Colab mostly for Kaggle competitions. So, I was very curious to see what the new subscription option would offer.

Two years ago, Google released Colab Pro. This was the first paid subscription option for Colab. For $9.99 per month, pro users get access to faster GPUs like the T4 and P100 if resources are available. Also, runtimes are longer in the pro version and instances are connected for up to 24 hours. In the free version, runtimes are limited to 12 hours and RAM is also limited to 16 GB. In the pro variant, it is possible to select a high-memory option and thus use 32 GB of RAM. The Google Pro+ variant now offers even more options to run Deep Learning relatively inexpensively without a cloud server or local machine. Let’s have a look [9].

2-2-1 The muscles — GPU and memory

Colab Pro+ offers access to the same GPUs as Colab Pro. However, Pro+ users are prioritized if the resources run short. In terms of RAM, Pro+ has seen another significant change.

While 32 GB of RAM is available in Colab Pro, Pro+ users have 52 GB available with the high-memory option.

That is about 1.6 times as much as pro users and 3.25 times as much as free users. This means that large amounts of data can also be processed outside of GPU applications [9].

2-2-2 Runtimes

The duration of the connection is very important in machine and deep learning. In the free variant, notebooks are disconnected after 12 hours, in the pro variant after 24 hours. What always bothered me a lot is the fact that the browser had to stay open, otherwise disconnects happened after about 90 minutes in my experience. This is for intensive testing ecologically not useful, because the whole time the local machine must remain on. And it happens every now and then that the browsers crash.

Here Pro+ offers from my point of view one of the most important innovations: **Background execution**. After the training has been triggered, the VM continues to run for up to 24 hours **without** the browser having to remain open.

The models and data are still stored in the workspace. I always push the data directly to my Google Drive, which makes background execution even more useful [9].

2-2-3 Availability

Unfortunately, nothing has changed in availability for now. Both Colab Pro and Colab Pro+ are available in the following countries: United States, Canada, Japan, Brazil, Germany, France, India, United Kingdom, and Thailand. Outside of these countries, Pro or Pro+ subscriptions are not available [9].

Table 3.2: Comparison of Colab Subscription plans [9].

	Colab Free	Colab Pro	Colab Pro +
Guarantee of resources	Low	High	Even Higher
GPU	K80	K80, T4 and P100	K80, T4 and P100
RAM	16 GB	32 GB	52 GB
Runtime	12 hours	24 hours	24 hours
Background execution	No	No	Yes
Costs	Free	9.99$ per month	49.99$ per month
Target group	Casual user	Regular user	Heavy user

2-3 Python language

Python is a high-level, interpreted, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.[10]

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a “batteries included” language due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0.[35] Python 2.0 was released in 2000 and introduced new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support. Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions. Python 2 was discontinued with version 2.7.18 in 2020.

Python consistently ranks as one of the most popular programming languages.

One of the most important uses of Python language is artificial intelligence.

2-4 Dataset

We used recently collected datasets (Vetrivel et al. 2018), highly varying in scene characteristics, lighting conditions, and image characteristics. These images were captured by different platforms and sensors from different earthquake events in several geographic locations. They are separated into two collections according to the underlying capture platform: Aircraft and UAVs. Table 3.3 summaries information about these datasets. Figure 3.1 shows samples of each dataset. The aircraft dataset contains 19,133 samples derived from images captured with five cameras (one nadir and four oblique views), with a spatial resolution around 10–16 cm. The damaged class of this dataset contains 9203 samples, while the non-damaged class 9930 samples. The UAV dataset includes 11,643 samples obtained from images captured at different heights, views, cameras and lighting conditions, with spatial resolutions ranging from 1 cm to 5 cm. The damaged and non-damaged classes of this UAV dataset have 5682 and 5961 samples, respectively [12].

Table 3.3: Description of the training and testing samples derived from images of two platforms for different geographic locations.

Platform	**Geographic location**	**Year of disaster event**	**Type of event**
Aircraft	Port-au-Prince, Haiti	2010	Earthquake
	Bidonville, Haiti	2010	Earthquake
	L'Aquila, Italy	2009	Earthquake
	Onna, Italy	2009	Earthquake
	Tempera, Italy	2009	Earthquake
	Mirabello, Italy	2012	Earthquake
UAVs	Ecuador, Peru	2016	Earthquake
	Kathmandu, Nepal	2015	Earthquake
	L'Aquila, Italy	2009	Earthquake
	Pingtung, Taiwan	2016	Earthquake
	Mirabello, Italy	2012	Earthquake
	Gronau, Germany	2013	Manually destructed industrial area

Figure 3.1: Some images from data set [12].

3 A previous study

Before we present our study, let us first present a previous study conducted on the same dataset [12]. In this previous related study, two set of experiments were conducted:

- ❖ The first set of experiments were designed to compare classification performances by using feature vectors extracted from the first fully connected layer only (FC1), from the second fully connected layer only (FC2), and from both layers (FC1&FC2). It should be noted that the use of FC2 feature vectors is a general case in several existing studies in the literature.

- ❖ The second set of experiments was conducted in order to investigate performance improvement by combining activation feature vectors extracted from different pretrained CNN models, as illustrated in Figure 3.2. The authors studied three combinations (hybrids) for FC1, FC2 and FC1&FC2 features, having the characteristics summarized in Table below.

Table 3.4: Studied hybrid feature representations [12].

Hybrid method	CNN models used	Combined FC1 or FC2 feature vector size	Combined FC1&FC2 feature vector size
Hybrid-1	AlexNet, VGG16 VGG19 and GoogLeNet	13 312	26 624
Hybrid-2	AlexNet, VGG16 and VGG19	12 288	24 576
Hybrid-3	VGG16 and VGG19	8192	16 384

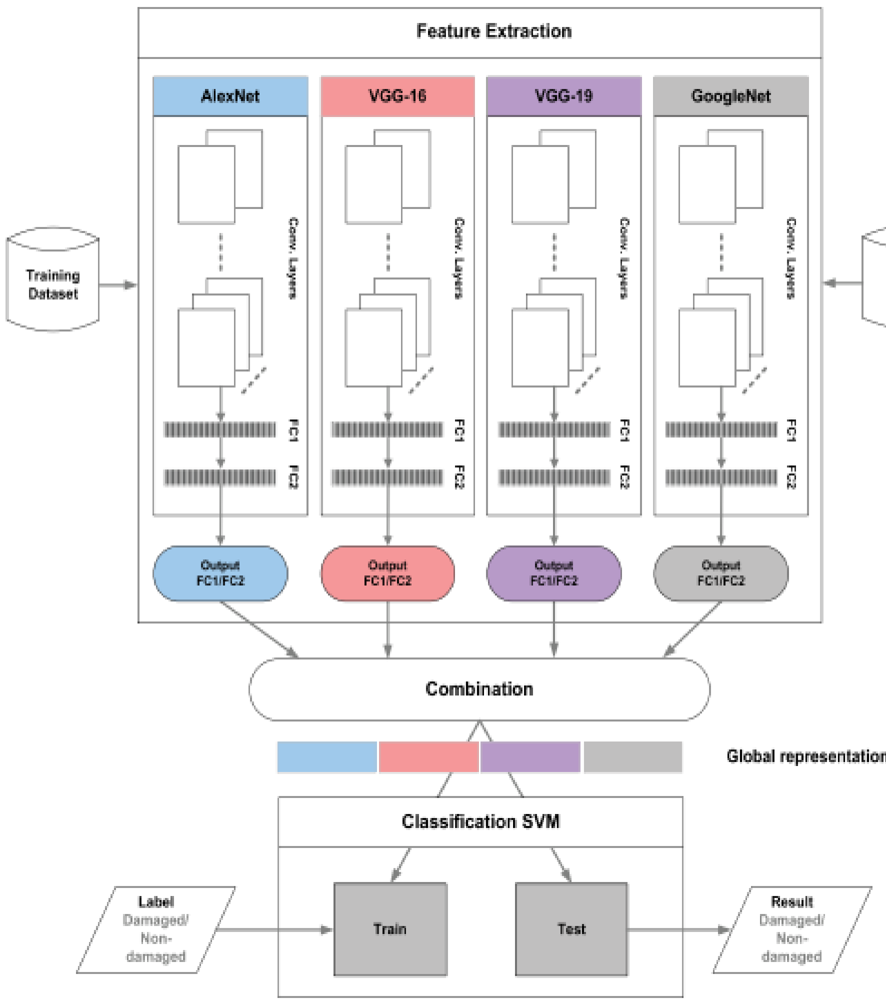

Figure 3.2: Overall schema proposed method for this study [12].

➢ **Combining features from different CNN models**

Table 3.5 gives the average of success rates (%) for the three hybrid methods listed in Table 3.4. The best result among the three hybrids for each feature nature is **boldfaced**. From statistical results using paired KW tests, there are highly significant differences between the performances provided by the hybrid representations and the performances related.

Table 3.5: Damage classification performance (SR) of hybrid feature representations [12].

CNN mode	Aerial dataset			UAV dataset		
	FC1	FC2	FC1&FC2	FC1	FC2	FC1&FC2
Hybird-1	97.51	97.44	**97.64**	97.62	97.62	**97.70**
Hybird-2	96.97	97.40	**97.61**	97.57	96.56	**97.67**
Hybird-3	97.02	97.12	**97.41**	97.37	96.26	**97.45**

4 Our proposal: Combining CNN and Gabor-based feature representations

Let's now move to explain our work in these steps:

1. Apply a pre-trained CNN model on the two datasets employed.
2. Apply Gabor filter to extract features and then apply a SVM algorithm to classify damage, using the same datasets.
3. Combine the CNN feature vector with Gabor features and apply a SVM classifier.
4. Study and compare the performance of three methods.

4-1 Why SVM

SVM is one of the supervised algorithms mostly used for classification problems. This article will give an idea about its advantages in general [17]**.**

- SVM is very helpful method if we don't have much idea about the data. It can be used for the data such as image, text, audio etc. It can be used for the data that is not regularly distributed and have unknown distribution**.**

- The SVM provides a very useful technique within it known as kernel and by the application of associated kernel function we can solve any complex problem. Kernel provides choosing a function which is not necessarily linear and can have different forms in terms of different data it operates on and thus is a non-parametric function.

4-2 Why Gabor filter

A Gabor filter (in image processing), named after Dennis Gabor, is a linear filter used for texture analysis, which means that it basically analyses whether there are any specific frequency content in the image in specific directions in a localized region around the point or region of analysis

Frequency and orientation representations of Gabor filters are claimed by many contemporary vision scientists to be similar to those of the human visual system, though there is no empirical evidence and no functional rationale to support the idea [18].

4-3 Deep learning (CNN Image classification using transfer learning, VGG-19)

The Convolutional Neural Network (CNN or ConvNet) is a subtype of the Neural Networks that is mainly used for applications in image and speech recognition. Its built-in convolutional layer reduces the high dimensionality of images without losing its information. That is why CNNs are especially suited for this use case.

Over the past few years, deep CNNs have revealed tremendous results in the area of computer vision. But still, the researchers are facing many challenges to execute the CNN model. The proposed system is implemented to resolve the issues that have arisen for the image classification task using a deep neural network.

The first challenge is to design a network model. CNN is designed with many layers, so they require millions of parameters to learn during the training phase. Designing a CNN model from the scratch demands a few resources for the execution, such as a large memory capacity, a fast processor, a huge dataset, enormous power consumption, etc. Deep learning needs an extremely large memory capacity as deep learning extracts a huge amount of data during the feature extraction phase. Basically, deep learning evaluates the value for each pixel of the image using various mathematical operations. Deep learning takes a lot of time for the computation (can be many hours or many days) depending on the computational capabilities of the hardware. So, power backup is required to make it a continuous process. Deep learning algorithms cannot be implemented on the general CPU system rather they need GPUs and TPUs enabled systems. These systems are very expensive and are not easily affordable. Deep learning works well with a large collection of data. The accuracy depends on the size of data which is very difficult to assemble in the real world. Even it makes the use of data augmentation to consider the various aspects of the image and to increase the size of the dataset, but still, it does not help to achieve the satisfactory results [13].

4-3-1 Feature extraction algorithms

Feature extraction plays the most significant role in image classification. The performance of the classification task highly depends on the crucial features of the images. The features of an object are classified into local and global features based on color, shape, or texture. Color and texture features are considered as local features and shape as global features. In this study, deep features and handcrafted features are extracted for image classification. Deep features are extracted from a pre-trained deep neural network (i.e., VGG19). The deep model extracts both the local and global features of an image. A brief description of each method is mentioned as follows [13].

4-3-2 Pre-trained CNN: VGG-19

VGG19 proposed by Simonyan and Zisserman (2014) is a convolutional neural network that comprises 19 layers with 16 convolution layers and 3 fully connected to classify the images into 1000 object categories. VGG19 is trained on the ImageNet database that contains a million images of 1000 categories. It is a very popular method for image classification due to the use of multiple 3×3 filters in each convolutional layer. The architecture of VGG19. This shows that 16 convolutional layers are used for feature extraction and the next 3 layers work for classification. The layers used for feature extraction are segregated into 5 groups where each group is followed by a max-pooling layer. An image of size 224 × 224 is inputted into this model and the model outputs the label of the object in the image. In the paper, features are extracted through a pre-trained VGG19 model, but for classification, various machine learning approach is followed. As the CNN model computes huge parameters after feature extraction, there is a need for dimensionality reduction to minimize the size of the feature vector.

The dimensionality reduction is done with Locality Preserving Projection that is followed by a classification method [13].

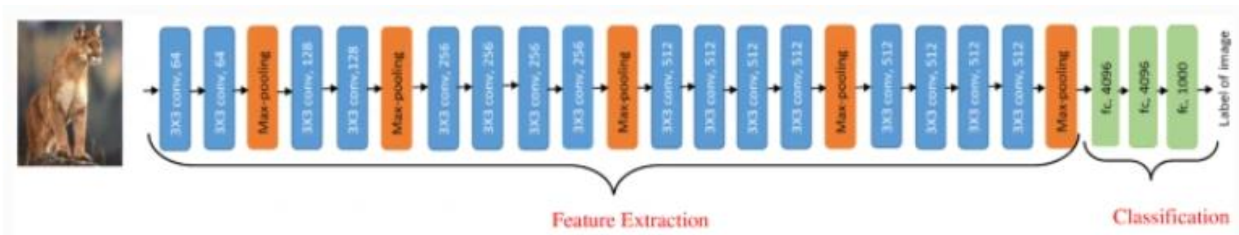

Figure 3.3: Architecture of VGG19 model [13].

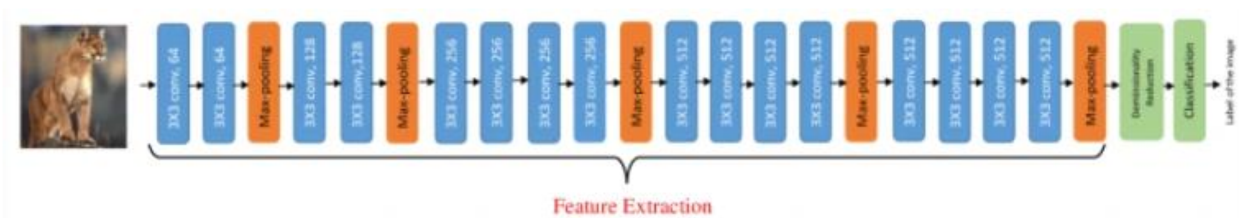

Figure 3.4: Ensemble of deep feature extraction using VGG19 model and machine learning classification [13].

After applying CNN using VGG19, we get results about loss and accuracy test:

```
[9]  results = model.evaluate(test_batches, batch_size=32, steps=test_batches.samples//32., verbose=1)

     190/190 [==============================] - 1151s 6s/step - loss: 0.3389 - accuracy: 0.8653
```

Figure 3.5: CNN Algorithm accuracy.

4-4 Machine learning (Gabor filter & SVM Algorithm)

A novel Support Vector Machine (SVM) damage classification method using optimized Gabor features is presented in this work. 25600 Gabor features are first selected by a boosting Algorithm, which are then combined with SVM to build a two-class based damage classification system. While computation and memory cost of the Gabor feature extraction process has been significantly reduced, our method has achieved the same accuracy as a Gabor feature and Linear Discriminant Analysis (LDA) based multi-class system**.**

But here we will use it for the purpose of earthquake classification.

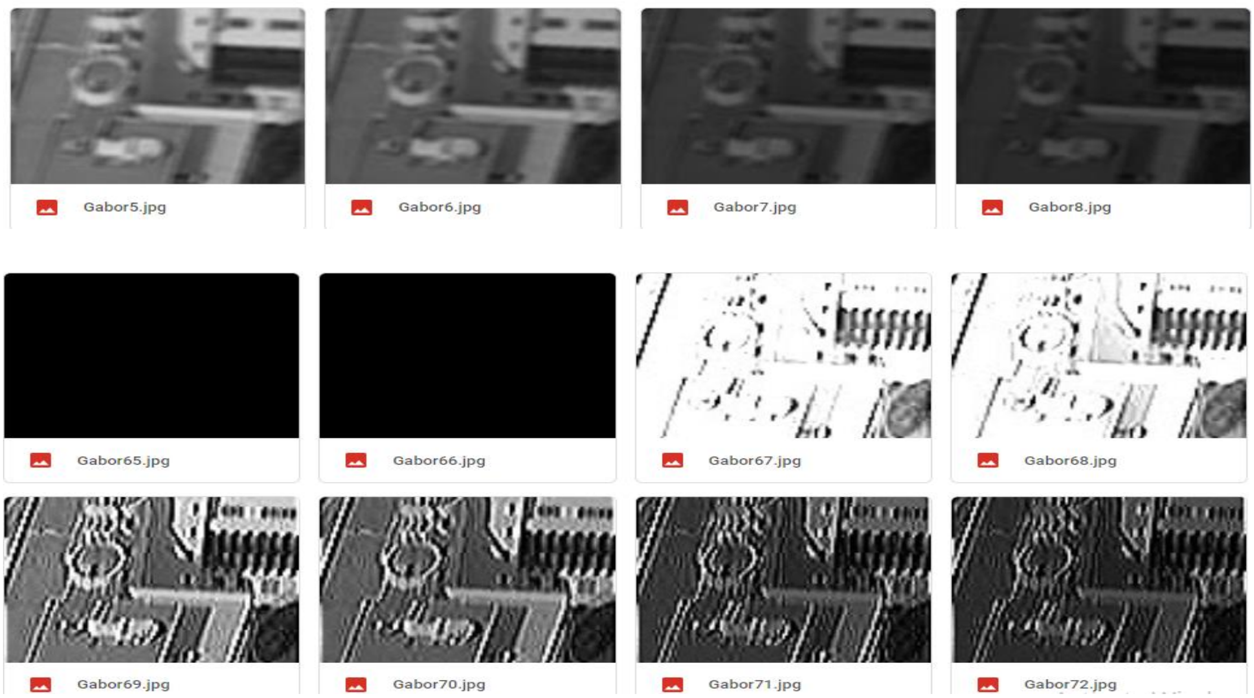

Figure 3.6: Some different Gabor filter applied on our datasets.

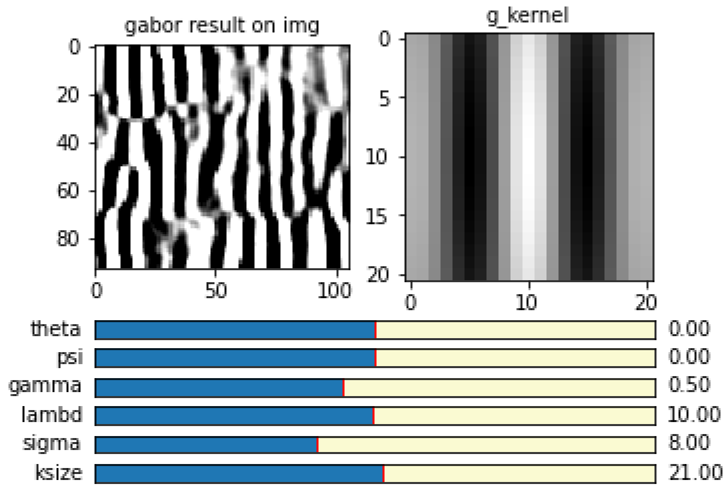

Figure 3.7: the best Gabor filter.

This means that the parameter in the filter Gabor is very important and has an impressive impact; with each different parameter we get different results.

After applying SVM using Gabor filters, we get these results:

	precision	recall	f1-score	support
No	0.86	0.80	0.83	800
Yes	0.81	0.87	0.84	800
accuracy			0.84	1600
macro avg	0.84	0.84	0.84	1600
weighted avg	0.84	0.84	0.84	1600

Figure 3.8: the Results of SVM using Gabor filters.

4-5 Combining CNN and Gabor features

We are going to take a last layer form the VGG19 model and merge it with Gabor features, then apply a SVM algorithm in **Figure 3.9** that display Diagram showing a summary of our work.

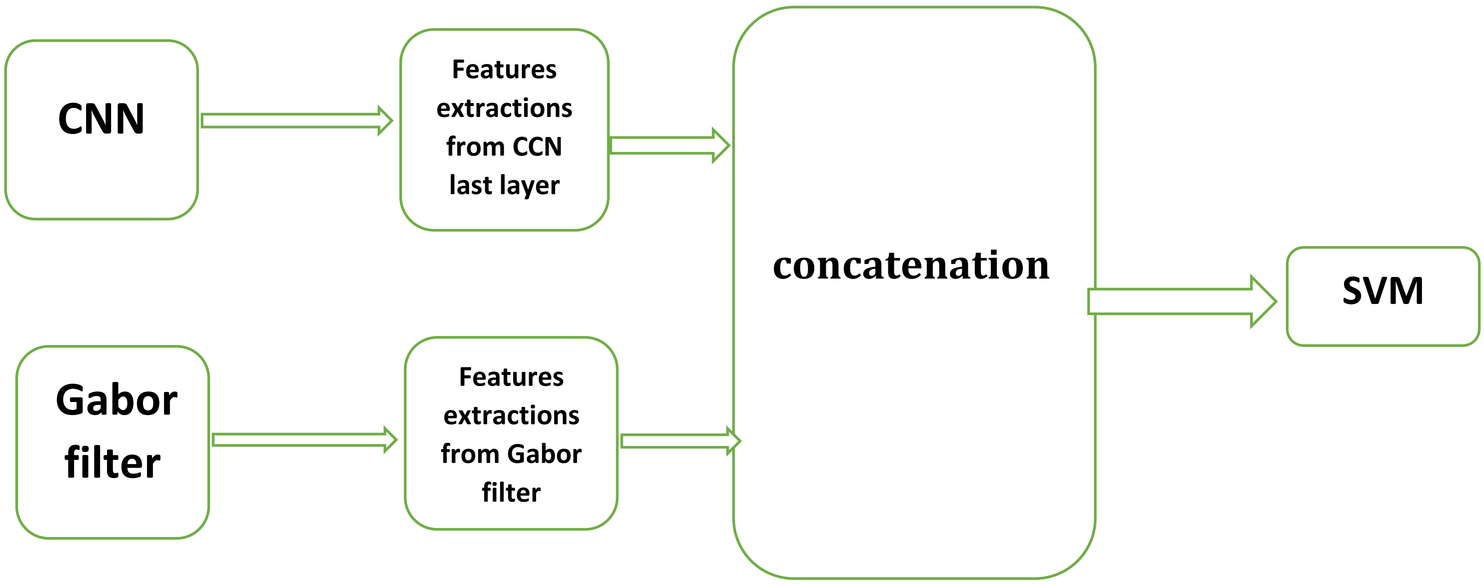

Figure 3.9: Diagram showing a summary of our work.

In all our experiments we have 8000 image in our dataset , we randomly selected 80% of each dataset as training samples and the other 20% as testing samples. Table 3.6 experiment settings to display number of train and number of test for each methods we used in this project , and in Table 3.7 lists the vector size different feature representations employed

Table 3.6: experiment settings.

CNN mode	UAV dataset		Areal dataset	
	number of train	number of test	number of train	number of test
cnn	6400	**1600**	6400	**1600**
Svm using gabor	6400	**1600**	6400	**1600**
concatenate	6400	**1600**	6400	**1600**

Table 3.7: Vector size of different feature representations studied.

CNN vector size	Gabor filter Vector size	Combined CNN && Gabor filter vector size
2048	25600	27648

4-6 Algorithm configuration

- ✓ CNN(VGG19)

The proposed method is applied with some configuration that makes the model train properly they are (Adam Optimizer,Learning rate equal to 1e-2, Batch of size 32,

loss = Binary Crossentropy, Categorical accuracy metric and finally val_acc and val_loss and save weights.)

- ✓ SVM

In the function LinearSVC we get this important parameter

(tol =1e-5 , C=1.0 , multi_class='ovr',   intercept_scaling=1, , verbose=0, random_state = None,   max_iter=1000**)**

- ✓ Gabor filter

This is the optimal Gabor parameters

Theta = 0 , psi = 0 , gamma = 0.50 , lambd = 10 , sigma = 8 , Size = 21.

The results we get through each method we used in our study are shown in Table 3.7.

Table 3.7: Comparison of different feature representations studied.

Method used / Dataset	CNN (VGG19)	SVM using Gabor	Concatenation of CNN last layer && Gabor feature with SVM
Aircraft	86%	82%	**96%**
UAVs	84%	84%	**97%**

4-7 Comparison with the study presented by Settou et al. [12]

Through the results obtained in the two studies (our study and the study in [12]) discussed in this chapter, which is to collect the mechanisms of machine learning and deep learning in different ways, where each method gives different results, but they are close in some cases.

The first study [12] extracted image features are from different pre-trained CNN models (AlexNet, VGG16 VGG19, and GoogLeNet), combined them, and passed all CNN features to the SVM algorithm. There was some improvement, but there was no great efficiency with respect to the use of one single pre-trained model. On the other hand, let's take into account that the combination of features extraction from fully connected layer 1 (FC1) and the fully connected layer 2 (FC2) is better than taking only one layer.

But our study is the simplest, easiest, and most efficient of extracting the features from the last layer of a pre-trained CNN model (VGG-19) and combining them with the features extracted using Gabor filters, and then passing all together to the SVM algorithm. In our study, the improvement amounted to more than 10% (see Table 3.7), but in [12] the improvement did not exceed 5%.

5 Conclusion

In the end, we conclude that the idea of combining machine learning and deep learning for damage classification is very feasible. According to our experiments conducted on different databases, our proposal can clearly and explicitly increase the classification performance.

General conclusion

Assessing building damage after an earthquake is a critical step in emergency response and recovery planning. The damaged state of a building can varies from superficial cracking to complete collapse depending on the building properties, soil conditions, and earthquake and ground motion characteristics. The spatial distribution of impacts in an earthquake-affected area makes building damage assessment a complex and time-consuming process. Tools such as ShakeMap and ShakeCast provide rapid assessments of earthquake damage. However, their ability to predict the spatial distribution of building damage with reasonable accuracy can vary depending on the methodology used for the assessment.

This study explores the effectiveness of using a hybrid ML and DL techniques to predict the earthquake damage to buildings. Because of the high risks of earthquakes and their continuous development, we need to develop more accurate and rapid algorithms in locating earthquakes and assessing damages for easy access to civil protection in a timely manner to save human lives as quickly as possible, and this is what we obtained in this study: Machine learning and deep learning to get a more accurate algorithm.

According to study that we have conducted for the damage classification using hybridization between ML and DL, we obtained results with high accuracy compared to each method separately:

- When using a standalone machine learning method, with Gabor filters and a SVM classifier, we get accuracy of 82% for Aircraft dataset (resp. of 84% for UAVs dataset).
- While when using the deep learning method alone, VGG-16, we get an accuracy of up to 82% for Aircraft dataset (resp. of 84% for UAVs dataset).
- But when using a hybrid between the two methods, a combined VGG-16 and Gabor feature vector and a SVM classifier, we get a high accuracy of 96% for Aircraft dataset (resp. of 97% for UAVs dataset).

This shows the importance of hybridization and its effectiveness in classifying earthquake damage, We can rely on it in the future and develop this technique, which is the hybridization technique, due to its success in improving accuracy, according to our study, so that it will surely lead to better results when adjusting the settings in a different way, such as using cross-validation and other techniques.

Appendix A

Features extraction using Gabor filter and CNN, then classification with SVM

```
def samples(dir):
  X = []
  Y = []
  for cls in os.listdir(dir):
    for img in tqdm(os.listdir(dir+cls)):
      #  (80.80.3) input image cnn for VGG19 model
      img = cv2.resize(cv2.imread(dir+cls+"/"+img),(80,80))

      #apply filter gry
      gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
      gray = cv2.resize(gray, (160,160))

      #normalization
      imgg = img/255.0

      #(1.80.80.3) add dimontion to input image svm
      imgg = np.expand_dims(imgg,axis=0)
      feature_extractor = Model(base.input,base.get_layer("block5_pool").output)
      features = feature_extractor.predict(imgg)
      res1 = features.reshape(np.prod(features.shape[1:]))
      # apply gabor filter
      gabor_kernel = cv2.getGaborKernel((6, 6), 10, 2*np.pi/360 * 25, 5, 0, 0, ktype=cv2.CV_32F)
      result = cv2.filter2D(gray, cv2.CV_8UC3, gabor_kernel)
      res2 = result.reshape(np.prod(result.shape))

      #concatinate between gabor feature and VGG19 feature
      res = np.concatenate((res1,res2),axis=0)

      X.append(res)
      if cls=="Yes":
        Y.append(1)
      else:
        Y.append(0)
  X, Y = np.array(X),np.array(Y)
  # returned feature
  return train_test_split(X,Y,random_state=5,test_size=0.2,stratify=Y)
```

Figure A.1: Feature extraction (using Gabor and CNN).

```
from sklearn.svm import LinearSVC
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
clf = make_pipeline(StandardScaler(),LinearSVC(random_state=None, tol=1e-5))
clf.fit(X_train, Y_train)
```

Figure A.2: SVM classification.

References

[1] https://viso.ai/computer-vision/image-classification/

[2] https://www.mygreatlearning.com/blog/feature-extraction-in-image-processing/

[3] https://www.analyticsvidhya.com/blog/2019/08/3-techniques-extract-features-from-image-data-machine-learning-python/

[4] https://sanchit2843.medium.com/introduction-to-machine-learning-and-deep-learning-bd25b792e488

[5] https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network

[6] https://levity.ai/blog/difference-machine-learning-deep-learning#:~:text=Machine%20learning%20means%20computers%20learning,as%20documents%2C%20images%20and%20text.

[7] https://www.analyticsvidhya.com/blog/2021/10/understanding-transfer-learning-for-deep-learning/#:~:text=For%20example%2C%20if%20you%20trained,understand%20the%20concepts%20in%20another.

[8] https://neurohive.io/en/popular-networks/vgg16/

[9] https://towardsdatascience.com/google-colab-pro-is-it-worth-49-99-c542770b8e56#:~:text=Colab%20Pro%2B%20offers%20access%20to,with%20the%20high%2Dmemory%20option.

[10] https://www.python.org/doc/essays/blurb/

[11] https://www.geeksforgeeks.org/support-vector-machine-algorithm/

[12] Tarablesse Settou, Mohamed-Khireddine Kholladi & Abdelkamel Ben Ali (2020): Improving damage classification via hybrid deep learning feature representations derived from post-earthquake aerial images, International Journal of Image and Data Fusion, DOI: 10.1080/19479832.2020.1864787

[13] Bansal, M., Kumar, M., Sachdeva, M., & Mittal, A. (2021). Transfer learning for image classification using VGG19: Caltech-101 image data set. Journal of Ambient Intelligence and Humanized Computing, 1-12.

[14] Copyright © 2022 FreedomVC | JetBlack by Firefly Themes

[15] Text is available under the Creative Commons Attribution-ShareAlike License 3.0; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.
[16] https://www.mygreatlearning.com/blog/deep-learning-applications/
[17] https://www.analyticsvidhya.com/blog/tag/svm/
[18] https://www.researchgate.net/post/Gabor_filter_and_its_applications
[19] www.potentiaco.com
[20] https://www.javatpoint.com