

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA

**Faculté des Mathématiques et de
l'Informatique**

Département d'Informatique

N° :.....



**DOMAINE : Mathématiques et
Informatique**

FILIERE : Informatique

OPTION : IA

**Mémoire présenté pour l'obtention
Du diplôme de Master Professionnelle
Par: BELKOFSI soheyb**

LEBOUAZDA manar

Intitulé

Classification et prédiction des plantes

Soutenu devant le jury composé de :

Dr. Barkat Abdelbassat	Université de M'sila	Président
Pr. Bourahla Mustapha	Université de M'sila	Rapporteur
Dr. Brahimi Belgacem	Université de M'sila	Examineur

Année universitaire : 2021 / 2022

DEDICACE

Aucune dédicace ne saurait exprimer mon respect, mon amour et ma considération pour les sacrifices que vous avez consentis pour mon instruction et mon bien être

Belkofsi soheyb

Lebouazda manar

REMERCIEMENTS

Nous sommes profondément redevables à notre directeur de thèse, le professeur BOURAHLA MUSTAPHA dont le soutien et l'inspiration indéfectibles ont fait de ce projet un grand succès. Nous la remercions de tout le soutien qu'il nous a apporté pour nous aider à réussir cette étude stimulante.

Un merci spécial à nos amis et à nos familles qui ont contenu les moments agités et le stress que nous avons vécus pendant le projet de recherche.

Lebouazda manar

Belkofsi soheyb

Table des matières

Liste De Figure	1
Introduction générale.....	1
L'intelligence artificielle	1
1.1 Définitions	2
1.1.1 Intelligence artificielle.....	2
1.1.2 Apprentissage machine (Machine learning).....	2
1.1.3 Apprentissage profond (Deep learning)	4
1.1.4 Généralité sur les images.....	5
1.1.5 Généralités sur les réseaux de neurones artificiels	6
1.1.6 Classification d'image par les réseaux de neurones convolutifs	7
1.1.7 Apprentissage par transfert avec les réseaux de neurones convo-lutifs	8
Les plants utilisé	9
2.1 Bougainvillea	11
2.1.1Description des Bougainvilliers	11
2.1.2 Besoins et mode de culture des Bougainvilliers.....	11
2.1.3 Entretien du Bougainvillea.....	12
2.1.4 Espèces et variétés de Bougainvillea.....	12
2.2 Marguerites de cap.....	13
2.2.1 Description des Marguerites de cap	13
2.2.2 Besoins et mode de culture des Marguerite de cap	13
2.2.3 Entretien de Marguerite de cap	14
2.2.4 Espèces et variétés de Osteospermum.....	14
2.3 Géranium zonal	15
2.3.1 Description du géranium zonal	15
2.3.2 Besoins et mode de culture des géraniums zonal	15
2.3.3 Entretien du géranium zonal	16
2.3.4 Espèces et variétés de gèranium zonal	16
2.4 Pètunia.....	17
2.4.1 Description du Pètunia	17
2.4.2 Besoins et mode de culture des Pétunias.....	17
2.4.3 Entretien du pétunia	18
2.4.4 Espèces et variétés de Pètunia.....	18
2.5 Dianthus	19
2.5.1 Description du Dianthus.....	19
2.5.2 Besoins et mode de culture Dianthus	19
2.5.3 Entretien du Dianthus.....	20

	vi
2.5.4 Espèces et variétés de Dianthus.....	20
2.6Gazania.....	21
2.6.1 Description du Gazania	21
2.6.2 Besoins et mode de culture Gazania.....	21
2.6.3 Entretien du Gazania	22
2.6.4 Espèces et variétés de Gazania.....	22
2.7 Aptenia Cordifolia.....	23
2.7.1 Dèscription du Aptenia Cordifolia	23
2.7.2 Besoins et mode de culture Aptenia Cordifolia.....	23
2.7.3 Entretien du Aptenia Cordifolia	24
2.7.4 Espèces et variétés d'Aptenia	24
2.8 Rosier de Chine	25
2.8.1 Description du Rosier de chine	25
2.8.2 Besoins et mode de culture Rosier de chine.....	26
2.8.3 Entretien du Rosier de chine	26
2.8.4 Espèces et variétés de Rosier de chine	26
2.9 Calendula Officinalis.....	27
2.9.1 Description du Calendula Officinalis	27
2.9.2 Besoins et mode de culture Calendula Officinalis	27
2.9.3 Entretien du Calendula Officinalis.....	28
2.9.4 Espèces et variétés de Calendula Officinalis.....	28
2.10 Aloe Maculata.....	29
2.10.1 Description du Aloe Maculata.....	29
2.10.2 Besoins et mode de culture Aloe Maculata	29
2.10.3 Entretien du Aloe Maculata.....	30
2.10.4 Espèces et variétés de Aloe	30
Implimentation	خطأ! الإشارة المرجعية غير معرّفة
3.1 Langage de programmation utilise (Python)	32
3.2 PyCharm Community Edition 2021.3.3	32
3.3 Les Bibliothèques utilisé	33
3.3.1 Tensorflow	33
3.3.2 Keras	33
3.3.3 NumPy.....	33
3.3.4 OS.....	33
3.3.5 Tkinter	34
3.3.6 Matplotlib.....	34
3.3.7 Random	34

	vii
3.3.8 Shutil	35
3.3.9 PIL (Python Imaging Library).....	35
3.4 Data processing.....	37
3.5 Create Model	41
3.6 Save Final Model	43
3.7 Evaluate Final Model.....	45
3. 8. L’interface graphique	46
3.9. Main.....	53
Conclusion et Perspective	54
BIBLIOGRAPHIE.....	55

Liste De Figure

Figure.2.1 :Réseau de neurone artificiel.....	1
Figure.2.1 :L'image de la plante Bouganvillea	12
Figure.2.2 :L'image de la planteMarguerit.....	14
Figure.2.3 :L'image de la plante G�ranium zonal.....	16
Figure.2.4 :L'image de la plante P�tunia.....	18
Figure.2.5 :L'image de la plante Dianthus.....	20
Figure.2.6 :L'imagede la plante Gazania.....	22
Figure.2.7 :L'image de la plante Aptenia Cordifolia	24
Figure.2.8 :L'image de la plante Rosier de chine	26
Figure.2.9 :L' image de la plante Calendula Officinalis.....	28
Figure.2.10 :L'image de la plante Maculata.....	30
Figure.3.1 :Les Biblioth�ques Python � Importes Dans La Page dataProcessin.....	37
Figure.3.2 :Les Biblioth�ques Python � Importes Dans La Page creat Model.....	37
Figure.3.3 :Les Biblioth�ques Python � Importes Dans La Page save Final Model.....	38
Figure.3.4 :Les Biblioth�ques Python � Importes Dans La Page evaluate Final Model.....	38
Figure.3.5 :Les Biblioth�ques Python � Importes Dans La Page de l'interface	38
Figure.3.6 :L' image dans datasat.....	40
Figure.3.7 :LeCodedelapage dataProcessing	42
Figure.3.8 :La fonction de charge le dataset.....	44
Figure.3.9 :La Fonction qui D�fini Le Model CNN.....	45
Figure.3.10 :La fonction Pour Trace Les Courbes est �valuer Un Mod�le.....	47
Figure.3.11 :Le Code De La Page Save Final Model.....	48
Figure.3.12 :Le Code De La Page Evaluate Final Mode.....	49
Figure.3.13 :L' image l'interface.....	50
Figure.3.14 ::L'image de l'interface	51
Figure.3.15 :L' image de l'interface.....	52

Figure.3.16:L'image de l'interface.....52

Figure.3.17:Le Code De La Page gui.....53

Figure.3.18:le perde d'entropie et la précision.....54

Figure.3.19:LeCode De La Page main.....55

Introduction générale

Définition des végétaux C'est un groupe d'organismes multicellulaires à noyau réel qui contiennent de l'ADN dans leurs cellules.

La présence de plantes est un élément très important de l'écosystème, où il y a environ 300 000 espèces de plantes à la surface de la terre, ce qui contribue au maintien de l'équilibre environnemental.

Les plantes produisent 98% de l'oxygène O₂ que les humains respirent et absorbent le CO₂, alors qu'elles travaillent sur la photosynthèse



ce qui aide à préserver l'atmosphère, à protéger la Terre des rayons ultraviolets et à réduire l'impact du réchauffement climatique.

Fournit 80% de nourriture pour les organismes vivants (humains, animaux).

Source de médicaments utilisés dans les soins de santé et divers parfums.

Il fournit également du bois dans les processus de construction ainsi que des matières premières pour la fabrication du papier.

Une partie du cycle de l'eau de la nature aide à transporter l'eau qu'elle absorbe du sol dans l'atmosphère par érosion.

En raison de leur grande diversité et de leur similitude entre ces plantes, de nombreux agriculteurs, amateurs de plantes et étudiants en biologie ont du mal à déterminer le type de plante et la famille à laquelle ils appartiennent et quels sont leurs avantages pour aller de l'avant avec la création d'un système informatique indépendant capable de détecter et de classer ces plantes avec une grande précision afin de faciliter leur mission.

Chapitre01

L'intelligence artificielle

Introduction

Nous commençons ce chapitre en définissant certains des sujets qui tournent autour de notre étude pour donner un aperçu général de tout ce qui concerne l'intelligence artificielle et les réseaux.

1.1 Définitions

1.1.1 Intelligence artificielle

L'intelligence artificielle (IA) est un ensemble de théories et de techniques permettant à des machines d'accomplir des tâches et de résoudre des problèmes normalement réservés aux humains et à certains animaux. En termes très simples, l'intelligence artificielle (IA) fait référence aux systèmes ou aux machines qui imitent l'intelligence humaine pour accomplir des tâches et qui peuvent s'améliorer itérativement en fonction des renseignements qu'ils recueillent. L'Intelligence Artificielle (IA) telle que nous la connaissons est une Intelligence Artificielle faible, par opposition à l'IA forte, qui n'existe pas encore. Aujourd'hui, les machines sont capables de reproduire un comportement humain, mais sans conscience. Plus tard, leurs capacités pourraient croître au point de se transformer en machines dotées de conscience, de sensibilité et d'esprit.

1.1.2 Apprentissage machine (Machine learning)

L'Apprentissage Machine ou Machine Learning est une technologie d'intelligence artificielle permettant aux ordinateurs d'apprendre sans avoir été programmés explicitement à cet effet. Cette technologie permet la reproduction de comportement grâce à des algorithmes, eux-mêmes alimentés par un grand nombre de données. Confronté à de nombreuses situations, l'algorithme apprend qu'elle est la décision à adopter et crée un modèle. La machine peut automatiser les tâches en fonction des situations.

Par exemple, pour qu'une machine apprenne le concept de chat, un ingénieur compile un grand nombre d'exemples d'images sur l'animal qu'il transmet à un algorithme. Auparavant, l'ingénieur devait établir la carte d'identité d'un chat (il a une fourrure, des moustaches, il retombe sur ses pattes, etc.) et représenter ces règles dans un programme informatique. Au-jour d'hui, il lui suffit de collecter les données, ce qui rend la tâche plus facile et plus rapide. Cette nouvelle façon d'automatiser conduit à des progrès considérables.

Apprentissage supervisé

L'apprentissage supervisé ou supervised learning est une approche d'apprentissage où les données sur lesquels le travail est effectué sont connus. On distingue des problèmes de régression (prédiction de variable continue quantitative), des problèmes de classification (prédiction de variable discrète qualitative). Par exemple, dans le cas d'un classificateur d'images, nous allons avoir un ensemble de données labellisées. Ainsi, nous connaissons l'étiquette de chaque image. À partir de ces étiquettes, nous allons pouvoir apprendre à notre modèle les différentes catégories représentées. Ainsi, à chaque prédiction de notre modèle, nous pouvons lui dire s'il a donné une bonne prédiction ou s'il s'est trompé. De ce fait, notre système apprend de ses erreurs. Il va chercher à minimiser l'erreur en fonction des données que nous allons lui donner.

Généralement, dans un apprentissage supervisé, nous avons trois grands temps. Un temps d'apprentissage, où nous allons donner à notre modèle un ensemble de données, où il pourra s'entraîner. Puis une phase de validation, où nous allons chercher à vérifier l'apprentissage de notre système. Pour ce faire, nous allons prendre un ensemble de données que nous n'avons jamais montré à notre modèle. Puis, nous allons chercher à vérifier si notre modèle arrive à bien différencier les différentes catégories. Enfin, nous avons une phase de production où nous allons pouvoir utiliser notre système.

Apprentissage non supervisé

Dans un apprentissage non supervisé ou un unsupervised learning, les catégories d'appartenance des données ne sont pas connues. Contrairement à l'apprentissage supervisé, nos données ne sont pas étiquetées. Ainsi, l'objectif va être, pour notre modèle, de repérer des similarités et, en fonction de celle-ci, de déterminer certaines données qui appartiennent à la même catégorie. Pour illustrer ce type d'apprentissage, nous avons un ensemble de données d'une population d'animaux. Nous avons récolté un ensemble de caractéristiques comme leur âge, leur poids, leur taille, etc. L'objectif va être de déterminer si certains animaux peuvent appartenir à la même espèce et ainsi les regrouper ensemble.

Nous mesurons la qualité d'une méthode de classification non supervisée par la capacité qu'a notre modèle de découvrir certains motifs cachés .

Apprentissage semi-supervisé

L'apprentissage semi-supervisé ou semi-supervised learning se situe entre le supervisé et le non-supervisé. En effet, dans ce type d'apprentissage, nous avons un ensemble de données étiquetées et non étiquetées. Ce qui est intéressant avec cette méthode est que nous laissons une certaine liberté à notre modèle. En effet, nous lui donnons des données sur lesquelles il va pouvoir extraire différentes caractéristiques et sur d'autres, il va devoir trouver des caractéristiques qui lui permettent de s'améliorer.

Ce type d'apprentissage est intéressant lorsque nous avons énormément de données. En effet, cela peut être fastidieux d'étiqueter l'intégralité de nos données. Cette méthode d'apprentissage nous permet d'éviter ce problème.

Apprentissage par renforcement

Dans le cas de l'apprentissage par renforcement, nous allons avoir un agent présent dans un environnement où il sera amené à prendre des décisions. Notre agent va se retrouver dans une série d'états où il aura à sa disposition un ensemble d'actions. L'objectif de notre agent est de prendre la meilleure décision possible. Pour définir la meilleure action, nous allons attribuer à chaque actions une récompense positive ou négative. Ainsi, l'objectif pour notre agent est d'obtenir la meilleure récompense possible.

Pour illustrer ce type d'apprentissage, nous pouvons prendre le cas d'un aspirateur autonome. L'environnement dans lequel il se situe est la pièce et son objectif est de la nettoyer. Pour ce faire, on peut attribuer pour chaque position de la pièce une récompense lui indiquant qu'il n'a pas encore nettoyé cette partie. Une fois passer sur une position, nous lui attribuons une récompense négative lui indiquant qu'il n'est pas censé repasser sur la même position. On pourrait aussi imaginer que lorsque notre robot aspirateur passe de l'état en charge à l'état à besoin d'être chargé, il faut qu'il privilégie le chemin de retour à la borne de chargement.

1.1.3 Apprentissage profond (Deep learning)

L'apprentissage profond ou deep learning est un sous-domaine du machine learning (apprentissage machine) .

Le Deep Learning est un apprentissage en profondeur. Il va chercher à comprendre des concepts avec davantage de précision, en analysant les données à un haut niveau d'abstraction. Comment ? Grâce à une compréhension non linéaire. Son fonctionnement s'apparente à celui du cerveau. Dans un réseau de neurones, des couches successives de données sont combinées pour apprendre les concepts. Les réseaux les plus simples ne présentent que deux couches : une d'entrée et une de sortie, sachant que chacune peut disposer de plusieurs centaines, milliers, voire millions de neurones. Plus elles augmentent, plus la capacité du réseau à apprendre des représentations de plus en plus abstraites se développe.

Pour illustrer le fonctionnement du Deep Learning, imaginez que les réseaux de neurones veulent apprendre à reconnaître les visages humains. Une première couche conçoit qu'il existe des pixels, les suivantes saisissent que plusieurs pixels forment un bord, et ainsi de suite, jusqu'à ce qu'elles acquièrent la notion de « visage ». Finalement, elles seront même capables de distinguer des visages spécifiques

.

1.1.4 Généralité sur les images

Une image est une représentation planaire d'un objet, d'une scène situé en général dans un espace tridimensionnel, elle est issue du contact des rayons lumineux provenant des objets formants la scène avec un capteur (caméra, scanner, rayons X, etc.). Il ne s'agit en réalité que d'une représentation spatiale de la lumière. L'image est considérée comme un ensemble de points auquel est affectée une grandeur physique (luminance, couleur). Ces grandeurs peuvent être continues (image analogique) ou bien discrètes (images digitales). Les différents types de format d'image sont :

.Image couleur RVB

L'œil humain analyse la couleur à l'aide de trois types de cellules photo 'les cônes'. Ces cellules sont sensibles aux basses, moyennes, ou hautes fréquences (rouge, vert, bleu). Pour représenter la couleur d'un pixel, il faut donc donner trois nombres, qui correspondent au dosage de trois couleurs de base : Rouge, Vert, Bleu. On peut ainsi représenter une image couleur par trois matrices chacune correspondant à une couleur de base ;

Image d'intensités

C'est une matrice dans laquelle chaque élément est un réel compris entre 0 (noir) et 1

(blanc). On parle aussi d'image en niveaux de gris, car les valeurs comprises entre 0 et 1 représentent les différents niveaux de gris ;

Image binaire

Une image binaire est une matrice rectangulaire dans l'élément valent 0 ou 1. Lorsque l'on visualise une telle image, les 0 sont représentés par du noir et les 1 par du blanc.

L'image est un ensemble structuré d'information caractérisé par les paramètres tels que : le pixel, la dimension et résolution, le voisinage, le niveau de gris, le contraste, la luminance, le bruit, le contour.

1.1.5 Généralités sur les réseaux de neurones artificiels

Un réseau de neurones artificiels ou Artificial Neural Network est un système informatique s'inspirant du fonctionnement du cerveau humain pour apprendre. En règle générale, un réseau de neurones repose sur un grand nombre de processeurs opérant en parallèle et organisés en tiers. Le premier tiers reçoit les entrées d'informations brutes, un peu comme les nerfs optiques de l'être humain lorsqu'il traite des signaux visuels. Par la suite, chaque tiers reçoit les sorties d'informations du tiers précédent. On retrouve le même processus chez l'Homme, lorsque les neurones reçoivent des signaux en provenance des neurones proches du nerf optique. Le dernier tiers, quant à lui, produit les résultats du système.

On distingue différents types de réseaux de neurones. Ils sont catégorisés en fonction du nombre d'épaisseurs qui séparent l'entrée de données de la production du résultat, en fonction du nombre de noeuds cachés du modèle, ou encore du nombre d'entrées et de sorties de chaque noeud. En fonction du type de réseau, la propagation des informations entre les différents tiers de neurones peut varier. Dans la variante la plus simple, celle du réseau de neurones dit "feedforward", les informations passent directement de l'entrée aux noeuds de traitement puis aux sorties. Les réseaux de neurones récurrents, quant à eux, sauvegardent les résultats produits par les noeuds de traitement et nourrissent le modèle à l'aide de ces résultats. Ce mode d'apprentissage est un peu plus complexe. Enfin, les réseaux de neurones convolutifs sont de plus en plus utilisés dans différents domaines : reconnaissance faciale, numérisation de texte, traitement naturel du langage.

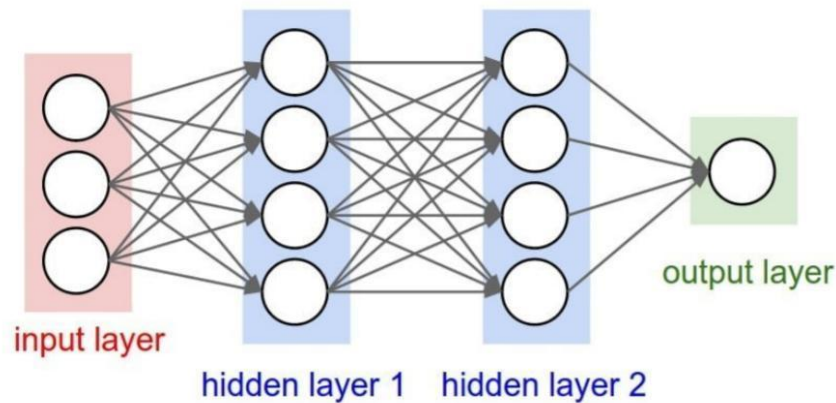


FIGURE 1.1 : Réseau de neurone artificiel

1.1.6 Classification d'image par les réseaux de neurones convolutifs

La classification automatique des images consiste à attribuer automatiquement une classe à une image à l'aide d'un système de classification. On retrouve ainsi la classification d'objets, de scènes, de textures, la reconnaissance de visages, d'empreintes digitale et de caractères. Il existe une multitude d'algorithmes pour la résolution des problèmes de classification parmi lesquels les CNN.

Les Réseaux de neurones convolutifs

Les réseaux de neurones convolutionnels sont à ce jour les modèles les plus performants pour classer des images. Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes. En entrée, une image est fournie sous la forme d'une matrice de pixels. Elle a 2 dimensions pour une image en niveaux de gris. La couleur est représentée par une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu].

La première partie d'un CNN est la partie convolutive à proprement parler. Elle fonctionne comme un extracteur de caractéristiques des images. Une image est passée à travers une succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions. Certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. Au final, les cartes de convolutions sont mises à plat et concaténées en

un vecteur de caractéristiques, appelé code **CNN**.

Dans le code **CNN** en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées (perceptron multicouche). Le rôle de cette partie est de combiner les caractéristiques du code **CNN** pour classer l'image. La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories. Une architecture **CNN** est formée par un empilement de couches de traitement indépendantes à savoir : la couche de convolution (CONV), la couche de pooling (POOL), la couche de correction ReLU, la couche entièrement connectée (Fully-Connected FC) et la couche de perte (LOSS)

1.1.7 Apprentissage par transfert avec les réseaux de neurones convo-lutifs

Le Transfer Learning (ou apprentissage par transfert) permet de faire du Deep Learning sans avoir besoin d'y passer un mois de calculs. Le principe est d'utiliser les connaissances acquises par un réseau de neurones lors de la résolution d'un problème afin d'en résoudre un autre plus ou moins similaire. On réalise ainsi un transfert de connaissances, d'où le nom. En plus d'accélérer l'entraînement du réseau, le Transfer Learning permet d'éviter le sur-apprentissage (overfitting). En effet, lorsque la collection d'images en entrée est petite, il est vivement déconseillé

d'entraîner le réseau de neurones en partant de zéro (c'est-à-dire avec une initialisation aléatoire) : le nombre de paramètres à apprendre étant largement supérieur au nombre d'images, le risque d'overfitting est énorme.

Conclusion

Dans ce chapitre, nous avons donné un aperçu de la nature de l'intelligence artificielle et des fonctionnalités qu'elle offre et de la façon dont les réseaux artificiels fonctionnent jusqu'à ce que nous parvenions à un ensemble de décisions à travers lesquelles nous développerons notre programme.

Chapitre 02

Les plantes utilisées

Introduction

Les plantes ornementales appartiennent au royaume botanique, situé dans la division des vases vasculaires, les fougères et les plantes ornementales sont connues comme le type de plante cultivée et développée pour être utilisée par le public. À de nombreuses fins telles que l'équipement des terrains de stade de sport, leur acquisition en tant que plantes durables et durables, les plantes ornementales sont utilisées dans de nombreux secteurs de l'industrie verte tels que les pépinières de plantes et de fleurs, les jardins, les bords de route et à des fins de traitement et de coordination des paysages. Leur entretien, bien que les plantes ornementales ne soient pas considérées comme une source de nourriture, d'énergie, de fibres et de médicaments, est utilisé dans de nombreuses questions importantes pour améliorer la qualité de vie, car elles sont utilisées comme pare-brise, fournissent de l'ombre, réduisent ou réduisent les facteurs d'érosion, purifient l'air et l'eau des polluants, y compris la poussière et les produits chimiques, et constituent un habitat et une source de nourriture pour les animaux sauvages.

La beauté est la caractéristique la plus importante de la plante ornementale et comprend la beauté de ses feuilles, de ses fruits, de son odeur, de la forme des jambes et de la barbe. L'humanité connaissait les plantes ornementales il y a 4 000 ans depuis le début des civilisations humaines.

2.1 Bougainvillea



Botanique

N. scientifique: *Bougainvillea x*

Famille : Nyctaginacées

Origine: Amérique du Sud

Floraison : printemps, été

Fleurs : blanc-crème, insignifiantes, entourées de bractées colorées (blanc, jaune, orange, rouge, violet)

Type : plante grimpante méditerranéenne et subtropicale

Végétation : vivace

Feuillage : persistant

Hauteur : 2 à 12 m

FIGURE 2.1 – L'image de la plant Bougainvillea

2.1.1 Description des Bougainvilliers

Les bougainvilliers hybrides sont des plantes hautes de 1 à 12 mètres, rampantes, sarmenteuses ou grimpantes, souvent exubérantes. Leur feuillage est normalement *persistant*, mais cette persistance peut être limitée par le froid. Leurs longues branches portent des feuilles alternes, ovales plus ou moins acuminées, longues de 4 à 13 cm. Ces branches portent aussi des épines, tendres d'abord et qui se recourbent en durcissant, créant une ancre pour s'accrocher à un support : arbre, mur, grillage... Ainsi ces plantes grimpantes ont besoin de peu de liens pour être palissées.

Les fleurs éclosent du printemps jusqu'en automne, parfois moins nombreuses en été. Les fleurs en elles-mêmes sont petites et tubulaires de couleur crème, elles sont assez insignifiantes, mais cependant nectarifères. Par contre, elles sont entourées de plusieurs grandes bractées papyracées de couleur vive. Leur aspect soyeux leur donne une teinte souvent très lumineuse : violette, lavande, rose, rouge, orange jaune ou blanche... parfois de couleur changeante.

2.1.2 Besoins et mode de culture des Bougainvilliers

Les bougainvilliers se cultivent dans une terre acide à presque neutre (un pH de 6,5 leur convient parfaitement) et qui est plutôt drainante. Ils sont assez résistants à la sécheresse, mais

ce qui les fait croître et fleurir abondamment est la chaleur et l'ensoleillement. Ils demandent à être ensoleillés dès le matin.

L'apparition des boutons floraux est stimulée par le soleil et les jours courts. Ils apparaissent quand les températures atteignent les 20 °C le jour et les 10 °C la nuit. Par contre, pendant les jours longs, un peu de sécheresse peut induire aussi la floraison.

2.1.3 Entretien du Bougainvillea

- Les bougainvilliers sont des plantes qui aiment les arrosages copieux, mais il ne faut pas non plus détrempier leurs racines.
- Supprimer les fleurs fanées au fur et à mesure de l'avancée dans la saison permet de stimuler sa généreuse floraison.
- Supprimez les mauvaises herbes

2.1.4 Espèces et variétés de Bougainvillea

à 18 espèces dans ce genre selon les classifications

- *Bougainvillea glabra*, bougainvillier glabre
- *Bougainvillea spectabilis*, bougainvillier remarquable
- *Bougainvillea peruviana*, une espèce moins dense avec des bractées arrondies
- *Bougainvillea arborea*, en arbre et avec des fleurs odorantes
- *Bougainvillea specto-glabra* 'Violet de Mèze' une variété connue pour sa bonne rusticité

2.2 Marguerites de cap



Botanique

N. scientifique : *Osteospermum*

Famille : Astéracées, Composées

Origine : Afrique du sud

Floraison : de mai aux gelées

Fleurs : blanc, rose, jaune, orange, violet

Type : fleur

Végétation : vivace selon les variétés, cultivé en annuelle

Feuillage : persistant

Hauteur : de 30 à 60 cm

FIGURE 3.4 – L'image de la plant Margurite

2.2.1 Description des Marguerites de cap

Les marguerites du Cap sont des plantes vivaces (bien que cultivées comme des annuelles en régions froides), qui comme leur nom l'indique, ressemblent fortement aux marguerites, mais sans en être. Selon les variétés, leur taille varient entre 30 et 60 cm. Les tiges souples se dressent et s'étalent, formant des touffes, voire des tapis pouvant atteindre 1 mètre à 1 mètre cinquante de large. Les nombreuses feuilles sont parfois simples et ovales, parfois découpées, et de couleur verte plus ou moins soutenue.

Les multiples fleurs apparaissent aux printemps. Ce sont de petit capitules qui, en fonction des variétés et des hybrides, se teintent de différentes couleurs, pales ou intenses : blanc, rose, jaune, orange, violet,... elles sont parfois bicolores.

2.2.2 Besoins et mode de culture des Marguerite de cap

Les Marguerite besoin d'un maximum de chaleur, de lumière et de soleil pour pousser, mais surtout pour s'épanouir et fleurir. Concernant la nature du sol, préférez une terre légère, sablonneuse, plutôt pauvre, mais surtout bien drainée. Ne l'installez pas en plein vent.

Dès le mois de mars, démarrez les semis de marguerites du Cap au chaud puis repiquez les jeunes plants en godets, que vous protégerez des gelées. A partir de la mi-mai, installez-les en place, en distançant les pieds de 30 cm.

2.2.3 Entretien de Marguerite de cap

- Il lui faut une exposition bien ensoleillée et un sol bien drainant (qui ne retient pas l'eau).
- Elle pousse sans entretien particulier. Arrosez simplement en cas de forte chaleur
- Supprimer les fleurs fanées au fur et à mesure de l'avancée dans la saison permet de stimuler sa généreuse floraison.
- apports d'engrais liquide riche en potasse et phosphore seront bienvenus durant l'été.

2.2.4 Espèces et variétés de Osteospermum

Le genre comprend environ 70 espèces

- Osteospermum 'Buttermilk' aux fleurs jaunes clairs, dressées
- Osteospermum 'Whirligig' aux pétales blancs, enroulés en forme de spatule, cœur bleu foncé
- Osteospermum Jamboana a un port dense qui fait de lui une plante adaptée aux petits jardin ou aux jardinières
- Osteospermum jucundum : fleurs mauve rosé

2.3 Géranium zonal



Botanique

N.scientifique : *Pelargonium x Hortorum*

Famille : Géraniacées

Origine : horticole

Floraison: de mai aux gelées

Fleurs : blanc, rose, rouge, saumon, bicolore

Type : fleur

Végétation : arbrisseau

Feuillage : persistant

Hauteur : 120 cm

FIGURE 2.3 :L'image de la plant Géranium zonal

2.3.1 Description du géranium zonal

Le géranium zonal forme un petit arbuste vivace dont les tiges se lignifient à la base. Il montre un port dressé, un peu ramifié, et porte des feuilles au limbe velouté, pelletées, presque ronde, souvent marqué d'un anneau rouge sombre. Ce feuillage persistant est aromatique et dégage une odeur forte lorsqu'on l'effleure. Les fleurs, assez grandes, de forme pleine et colorées de blanc, rouge, vif, rose mauve ou saumon ; elles se développent à partir de l'aisselle des feuilles, du printemps jusqu'aux gelées. Elles sont regroupées en panicules bien rondes. Ces fleurs sont fécondées par les insectes et produisent des graines si l'on ne coupe pas les têtes défleuries.

2.3.2 Besoins et mode de culture des géraniums zonal

Le géranium zonal n'est pas rustique, il s'abime avec les gelées, et ne croît qu'au-dessus de 13 °C, c'est pourquoi on ne les achète au printemps que lorsque tout risque de gelées est écarté.

Le géranium zonal se cultive bien en jardinière, potée, mais aussi en pleine terre le temps d'une saison. Il demande une terre riche et drainante. Il peut être installé dans un mélange pour géranium, ou dans un mélange de 2/3 terre franche de jardin et de 1/3 terreau bien décomposé. Ce dernier mélange est plus lourd en masse, mais retient l'eau d'arrosage plus longtemps.

Le géranium zonal apprécie la chaleur et le soleil direct. Il est donc exposé au soleil du sud ou sud-ouest. Il est très résistant à la sécheresse, mais grandit et fleurit bien davantage avec

des arrosages réguliers. Il faut donc arroser vos jardinières de géranium à chaque fois que la surface du substrat est sèche sur 1 ou 2 centimètres.

2.3.3 Entretien du géranium zonal

- Les jeunes plants ont parfois besoin d'être pincés pour favoriser les ramifications.
- Gourmand, le géranium zonal sera nourri d'un apport d'engrais liquide, riche en phosphate et en potasse, tous les 10 jours, ou d'engrais à libération lente tous les 3 mois.
- Les têtes défleuries sont ôtées au fur et à mesure pour qu'il refleurisse davantage

2.3.4 Espèces et variétés de géranium zonal

Nombreuses variétés horticoles à choisir fleuries

- Géranium 'Princess of Wales' aux fleurs très doubles rouges et blanches
- Géranium 'Panrou' aux fleurs rouges et blanches mouchetées

2.4 Pétunia



Botanique

N. scientifique : *Petunia x*

Famille: Solanacées

Origine : horticole

Floraison : fin printemps, été, automne

Fleurs : blanc, rose, rouge, violet

Type : fleur

Végétation: vivace cultivé comme une annuelle

Feuillage : persistant

Hauteur : 30 cm

FIGURE 2.4 : L'image de la plante Petunia

2.4.1 Description du Pétunia

La caractéristique des pétunias est leur floraison en forme de trompette dans des couleurs blanches, roses, violettes, rouges ou bleues, avec des motifs tachetés, étoilés ou rayés, qui apparaissent sans relâche et en grand nombre de mai à octobre. En ce qui concerne la taille des fleurs, les pétunias Grandiflora et les pétunias Multiflora se distinguent. Les représentants des cultivars Grandiflora se distinguent par une très grande fleur plate, individuelle sur la tige. Les pétunias Multiflora, en revanche, portent des fleurs un peu plus petites, serrées les unes contre les autres, et leurs pousses plus courtes leur donnent un aspect plutôt compact. Les spécimens doubles rappellent les pétales de rose avec leurs têtes rondes et sont donc parfois commercialisés sous le nom de pétunias « roses ».

Les feuilles du pétunia sont vert foncé, à bords lisses et légèrement velues. Typiquement, elles collent légèrement.

2.4.2 Besoins et mode de culture des Pétunias

Les pétunia se cultivent dans une terre neutre et qui est plutôt drainante. elle est cultivée bien en jardinière, potée, mais aussi en pleine terre le temps d'une saison, besoin d'un maximum de chaleur, de lumière et de soleil pour pousser, mais surtout pour s'épanouir et fleurir.

Les semis de pétunias se font de février à mars, en terrine et au chaud. Repiquez les plants en godet, au stade de 3 ou 4 vraies feuilles, et gardez-les à l'abri. Mi-mai, mettez-les en terre, en potée ou en jardinière.

2.4.3 Entretien du pétunia

Un fois bien installé au jardin ou dans sa jardinière, le pétunia est très facile d'entretien.

- Retirez simplement les fleurs fanées au fur et à mesure afin de favoriser l'apparition de nouvelles fleurs.
- Supprimez les mauvaises herbes qui poussent au pied du pétunia.

2.4.4 Espèces et variétés de Pétunia

Le genre comprend environ 40 espèces Variétés classées suivant des séries et suivant la taille des fleurs :

- Pétunia série Grandiflora aux grandes fleurs
- Pétunia série Prélude aux fleurs de 20 cm
- Pétunia série Fangio aux fleurs de 20 cm
- Pétunia série Cascade au port retombant et à grandes fleurs
- Pétunia série Fanfare aux fleurs doubles
- Petunia série Star aux fleurs de 10 cm avec une bande blanche au centre des pétales
- Pétunia série Multiflora aux fleurs moyennes

2.5 Dianthus



Botanique

N. scientifique : *Dianthus plumarius*

Famille : Caryophyllacées

Origine : Europe centrale, Orient

Floraison: mai à juillet

Fleurs : jaune, rouge, blanc, orange

Type : fleur

Végétation : vivace

Feuillage : persistant

Hauteur : 30 cm

FIGURE 2.5 : L'image de la plante Dianthus

2.5.1 Description du Dianthus

L'œillet, qu'il soit de type ancien ou moderne, fait partie du vaste genre des Dianthus originaires des zones tempérées d'Asie, Europe et d'Afrique du Sud. Il se développe en coussin arrondi très compact formé par des tiges dressées portant un feuillage linéaire gris/vert d'où émergent des fleurs de 3 à 6 cm de diamètre se déclinant du blanc pur au rouge éclatant pour les variétés unicolores. Il existe bien sûr des cultivars bicolores, très doubles ou simples dont les pétales peuvent être délicatement arrondis, ou très frangés. Les Dianthus de type ancien, fleurissent vers le mois de juin sur une courte période, les cultivars modernes permettent de palier à ce petit défaut et offrent profusion de fleurs de juin à octobre.

2.5.2 Besoins et mode de culture Dianthus

Dianthus ne sont pas des plantes de culture difficile. Vivaces et rustiques une fois bien installés, ce sont des sujets sans souci. Quelques conditions doivent toutefois être réunies pour leur bien être : un emplacement très ensoleillé est indispensable ainsi qu'une terre bien drainée et légère. Évitez de planter ces œillets en terre constamment détrempée ou argileuse car ils seraient bien vite atteints par la rouille ou certaines viroses et finalement ne survivraient pas.

Un sol neutre ou même légèrement acide convient bien à ces espèces, qui, contrairement aux autres œillets aimant le calcaire, tolèrent une légère acidité.

2.5.3 Entretien du Dianthus

Arrosez pour une culture en pot. En pleine terre, arrosez uniquement en cas de sécheresse.

- Protégez les jeunes plants des escargots et des limaces.
- Coupez les fleurs fanées.

2.5.4 Espèces et variétés de Dianthus

Le genre comprend plus de 300 espèces

- *Dianthus plumarius* 'Gitane' .
- *Dianthus barbatus* .
- *Dianthus caryophyllus*.
- *Dianthus cathusianorum*.

2.6Gazania



Botanique

N. scientifique : *Gazania*

Famille : Astéracées, Composées

Origine : Afrique du sud

Floraison: de juin à octobre

Fleurs :jaune, orange, rouge, brun, rose

Type : fleur

Végétation: annuelle ou vivace selon l'espèce

Feuillage : persistant

Hauteur : 30 cm

FIGURE 2.6 : L'image de la plante Gazania

2.6.1 Description du Gazania

Le gazania est une plante vivace souvent cultivée en annuelle dans les régions fraîches qui fait partie de la famille des Astéracées. Le feuillage vert brillant est souvent duveteux à son revers, il forme des touffes importantes qui peuvent s'étendre avec les années si le climat le permet. Dès le mois de juin, de grosses fleurs solitaires apparaissent s'ouvrant chaque jour sous les rayons du soleil. Ressemblant à de grosses marguerites les coloris souvent très chauds et lumineux se déclinent du blanc au rouge vif en passant par toutes les nuances de jaune et d'orange. Les pétales peuvent être bicolores voire tricolores, souvent marginés d'une nuance plus claire, sur certaines variétés, leur base souvent plus sombre, forme un cercle très graphique autour du cœur de la fleur.

2.6.2 Besoins et mode de culture Gazania

Une plante idéale dans les jardins ensoleillés ,Le gazania est une plante de choix pour le jardinier qui a peu de temps à consacrer à l'entretien de son jardin. Il n'est pas difficile et se contente de tous les types de terres bien drainées et supporte même les sols sablonneux et les embruns. Il adore la chaleur et le plein soleil, auxquels il offre toute la beauté de sa floraison de longue durée qui peut s'étaler de la fin mai au début octobre. Une fois bien installée, cette plante n'a pratiquement pas besoin d'arrosage, Le gazania sera cultivé comme une annuelle dans la majorité des régions car il ne supporte pas le gel excepté sur une très courte période. Sur le pourtour méditerranéen et dans les zones abritées, il pourra cependant vivre de nombreuses années et former des touffes imposantes qu'il sera possible de diviser au printemps.

2.6.3 Entretien du Gazania

Gazania fleur L'entretien du gazania est quasi nul mais la floraison sera

d'autant plus belle si vous :

- Plantez dans du terreau pour plantes fleuries
- Supprime les fleurs fanées au fur et à mesure afin de ne pas épuiser la plante et stimuler l'apparition de nouvelles fleurs.
- Arrosez de temps en temps vos gazanias si ils sont en pot
- Apport possible d'engrais pour plantes fleuries en cours de saison

2.6.4 Espèces et variétés de Gazania

Le genre comprend 15 espèces :

- Gazania 'Gazoo' aux grandes fleurs jaunes, oranges ou rouges qui a la particularité de s'épanouir dans des conditions plus faibles d'éclairement que les Gazanias
- Gazania 'Talent jaune' vivace aux fleurs jaunes

2.7 Aptenia Cordifolia



Botanique

N. scientifique : *Aptenia cordifolia*

Synonymes : *Litocarpus cordifolia*,
Mesembryanthemum cordifolia

Famille : Aizoacées

Origine : Afrique du sud

Floraison : du printemps au début de l'automne

Fleurs : blanc, rose, rouge

Type : vivace

Végétation : vivace

Feuillage: feuillage succulent persistant

Hauteur : 5 cm

FIGURE 3.4 : L'image de la plante *Aptenia Cordifolia*

2.7.1 Dèscription du *Aptenia Cordifolia*

L'Apténie cordée ou bien *Aptenia Cordifolia* est une petite plante vivace au feuillage succulent. Elle fait partie de la famille des Aizoacées et peut parfois être assimilée à un ficoïde. Ses petites feuilles bien charnues d'un beau vert vif brillant au soleil sont portées par de fines tiges pouvant s'étaler de manière spectaculaire formant de véritables tapis. De l'été à l'automne, apparaissent de petites fleurs rouges de 1 à 2 cm de diamètre en forme de marguerites. Certains cultivars assez rares dans le commerce, présentent des fleurs blanches ou roses. D'une hauteur maximale de 5 cm, la plante peut toutefois s'étaler sur plusieurs mètres habillant délicatement buttes et murets. Idéale en rocailles, elle peut aussi être cultivée en potées au balcon ou en paniers suspendus dans une serre.

2.7.2 Besoins et mode de culture *Aptenia Cordifolia*

L' Apténie cordée ne lui confèrent pas une grande *rusticité* ; on privilégiera donc sa plantation dans les régions où le climat est clément et où les températures descendent rarement en dessous de 0°C.

Plantez l'Apténie en mai, lorsque tout risque de gel. Ne s'entend pas avec un sol humide ,En pleine terre, le sol doit être très drainé. Un sol sablonneux, pierreux, pauvre lui convient très bien .Les fleurs ont besoin du plein soleil pour s'épanouir. Cette plante peut toutefois être

cultivée à la mi-ombre dans les régions les plus chaudes, mais évitez l'ombre totale qui supprimerait la floraison .

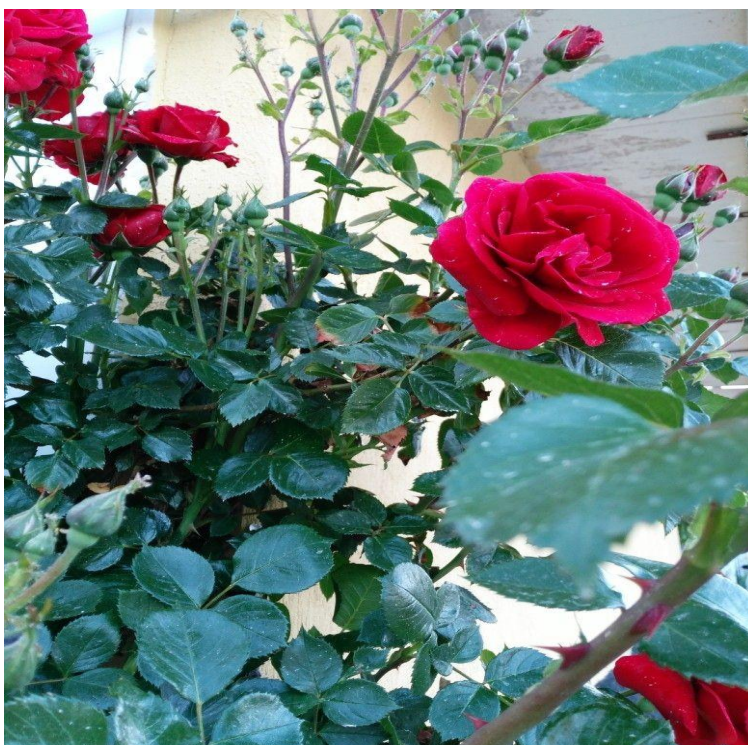
2.7.3 Entretien du Aptenia Cordifolia

- L'aptènia cordifolia demandent peu de chose : un peu d'eau de temps en temps pour palier à des séchesses trop longues.
- Pour éviter les semis spontanés, vous devrez supprimer les fleurs fanées avant qu'elles forment des fruits contenant les graines.

2.7.4 Espèces et variétés d'Aptenia

- *Aptenia lancifolia* : sous-arbrisseau au feuillage en forme de lance. Fleurs rose violacé. 10 cm de hauteur.
- *Aptenia haeckeliana* : feuillage ovale et lancéolé vert olive à jaunâtre. Fleurs jaunes ou rouges. 5 cm de hauteur.
- *Aptenia geniculiflora* : sous-arbrisseau au feuillage cylindrique vert pâle. Fleurs blanc crème. 10 cm de hauteur.
- *Aptenia cordifolia* 'Variegata' : feuillage vert marginé de blanc. Fleurs mauve foncé. 5 cm de hauteur.
- *Aptenia cordifolia* 'Sunny Sue' : feuillage vert foncé, fleurs blanches. 10 cm de hauteur.

2.8 Rosier de Chine



Botanique

N. scientifique : *Rosa chinensis*

Synonymes: *Rosa indica*, *Rosa chinensis var spontanea*

Famille : Rosacées

Origine : Chine

Floraison: de juin jusqu'aux gelées

Fleurs : rose , rouge, blanc, orange, jeune

Type : rosier botanique

Végétation : arbustive

Feuillage : caduc

Hauteur : 1m 30 à 2 m

FIGURE 2.8 : L'image de la plante Rosier de Chine

2.8.1 Description du Rosier de chine

Rosa x centifolia est un rosier haut de 1,50 m à 1,80 m aux tiges dressées puis retombantes, que l'on peut cultiver comme un rosier arbustif de grande taille ou palissé comme un rosier semi-grimpant. Ses branches sont assez souples et plutôt fines à leur extrémité, portant de petits aiguillons. Son feuillage est souvent sain, plus ou moins dense. Les feuilles sont composées de 5/7 folioles vert bleuté, bordé d'un fin liseré rouge.

Il n'est pas remontant, il fleurit entre mai et juillet, soit par rose solitaire soit par corymbe de plusieurs roses.

La rose possède un très grand nombre de sépales pétaloïdes, serrés les uns contre les autres cerclés de pétales un peu plus longs ; ces derniers dessinent un cercle presque parfait autour d'un large cœur très dense. L'ensemble est très globuleux en début de floraison, puis s'élargit en plateau.

2.8.2 Besoins et mode de culture Rosier de chine

Loin d'être délicat, *Rosa indica* est vigoureux et résistant aux maladies. Il ne demande pas de nature de sol particulière, et est tolérant pour le pH. Son exposition est préférentiellement très ensoleillée, car c'est un gage d'une importante floraison, il supporte néanmoins la mi-ombre. Bien que parfois accusé de frilosité, il résiste à de fortes gelées. Dans les régions les plus gélives, il sera de préférence installé au jardin au printemps. Pralinage et paillage à la transplantation, plus quelques arrosages lui permettent de s'enraciner en profondeur dès sa première saison. Une fois bien repris il supporte plutôt bien la sécheresse.

2.8.3 Entretien du Rosier de chine

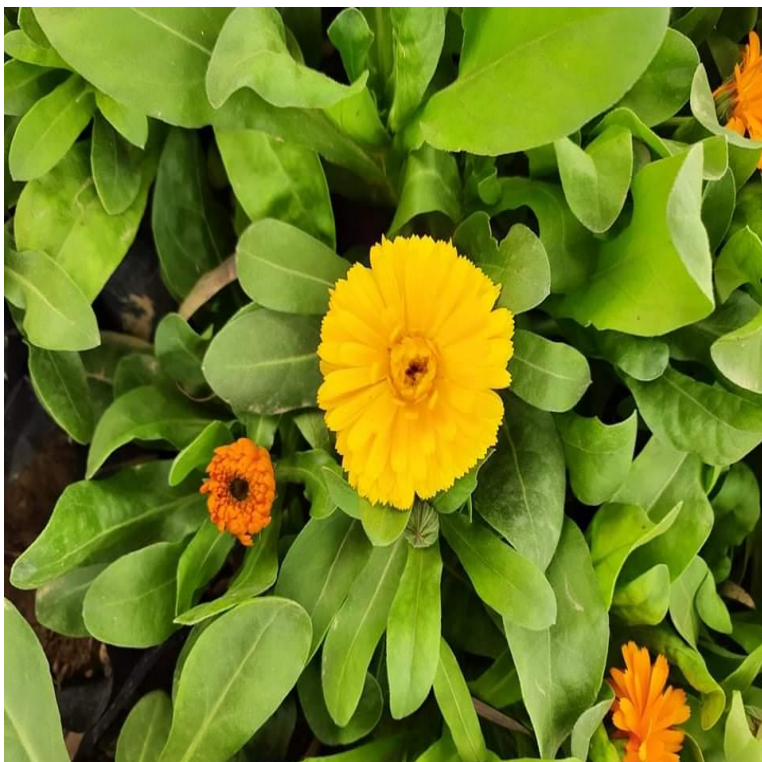
- S'il y a besoin de limiter son encombrement ou de nettoyer les tiges défleuries, la taille intervient en fin d'hiver, juste avant le redémarrage de la croissance.
- Les tiges les plus anciennes sont éventuellement rabattues à cette même époque, pour permettre un renouvellement. Le bois mort est nettoyé.
- Il n'y a pas besoin d'ôter les fleurs fanées pour que le rosier de Chine poursuive sa floraison.

2.8.4 Espèces et variétés de Rosier de chine

150 espèces et des milliers de cultivars

- *Rosa banksia*, le rosier de Banks.
- *Rosa rugosa*, à gros fruits.
- *Rosa gigantea*, de 30 m de hauteur parfois.
- *Rosa rubiginosa*, l'églantier rouge.
- *Rosa moschata*, le rosier musqué .
- *Rosa foetida*, le rosier fétide.
- *Rosa wichuraiana*, le rosier de Wichura.
- *Rosa rubiginosa*, l'églantier rouge.
- *Rosa hugonis*, aux fleurs jaunes.
- *Rosa chinensis* 'Mutabilis', superbe variété aux fleurs changeantes.
- *Rosa chinensis* 'Viridiflora', une curiosité aux fleurs vertes.
- *Rosa chinensis* 'Semperflorens', les rosiers du Bengale.
- *Rosa chinensis* 'Minima', 'Rouletti' ou 'Pumila', des formes naines.
- *Rosa chinensis* 'Sanguinea', fleurs rouges et simples.

2.9 Calendula Officinalis



Botanique

N. scientifique : *Calendula officinalis*

Famille : Astéracées, Composées

Origine : Europe

Floraison : d'avril aux gelées

Fleurs : jaune, orange

Type : fleur

Végétation : annuel

Feuillage : caduc

Hauteur : jusqu'à 70 cm, selon les variétés

Toxicité : les fleurs et les boutons floraux sont comestibles

FIGURE 2.9 : L'image de la plante Calendula Officinalis

2.9.1 Description du Calendula Officinalis

Calendula officinalis, ou bien Le souci est une plante annuelle quasiment rustique, qui est utilisée en massifs ou en fleurs coupées. Il se ressème tout seul au gré du vent.

Le souci forme une touffe dressée buissonnante, un peu étalée, avec des feuilles aromatiques, alternes, lancéolées, vert clair, duveteuses, mesurant de 10 à 15cm de long.

Les fleurs qui se succèdent de mai à octobre ressemblent à de grosses marguerites jaunes, simples ou doubles, atteignant 10 cm de diamètre. Les coloris sont variés : orange, jaune, abricot, crème, avec le plus souvent le fleuron, au centre, plus sombre.

Il possède de très nombreuses vertus médicinales ; c'est un antiviral, anti-tumoral, anti-inflammatoire, anti-œdémateux et antioxydant. Il est aussi utilisé en cosmétique.

2.9.2 Besoins et mode de culture Calendula Officinalis

Le souci n'est pas difficile à cultiver. *Calendula officinalis* tolère tous types de sols pourvu qu'il se trouve exposé au soleil pas nécessairement brûlant, ou à la mi-ombre.

Le souci doit être semé en place, à partir d'avril jusqu'au mois de juin pour échelonner les floraisons. Un éclaircissage des plantules sera alors nécessaire pour ne garder qu'un plant vigoureux tous les 10 à 15 cm.

Une fois bien installé, il n'est pas très gourmand en eau et peut se contenter d'un arrosage copieux par semaine en pleine terre si la chaleur n'est pas écrasante

2.9.3 Entretien du Calendula Officinalis

- Supprimez les fleurs fanées pour faciliter la remontée.
- Pincez l'extrémité de la plante pour que des tiges latérales se forment.
- Peu gourmande en eau, mieux vaut ne pas mouiller le feuillage pour éviter une attaque d'oïdium.

2.9.4 Espèces et variétés de Calendula Officinalis

Le genre comprend une vingtaine d'espèces

- Calendula officinalis 'Baby orange'aux fleurs doubles

2.10 Aloe Maculata



Botanique

N. scientifique : *Aloe maculata*

Synonymes : *Aloe saponaria*, *Aoe picta*, *Aloe maculosa*, *Aloe latifolia*

Famille : Asphodélacées

Origine: Afrique du Sud

Floraison: avril, mai, voit n'importe quand

Fleurs : rouge à orange

Type : succulente

Végétation : vivace

Feuillage: persistant

Hauteur : jusqu'à 70 cm

FIGURE 3.4 : L'image de la plante Aloe Maculata

2.10.1 Description du Aloe Maculata

Aloe maculata est une plante succulente très charnue. Ses rosettes de feuilles sont régulières et sans tiges ou avec l'âge, portées par de courtes tiges. Ses feuilles épaisses et larges sont globalement triangulaires, voire galbées. Elles sont aplaties, lisses et marginées de dents courtes. Elles sont de couleur olivâtre à mi-ombre et tirant sur le rose violacée au soleil, maculée de taches plus claires : de petites taches ovales, alignées plus ou moins en stries transversales. La pointe des feuilles sèche et se replie lorsque la plante fait face à une sécheresse prolongée.

Aloe maculata produit une hampe florale qui sort à proximité de l'apex. Elle s'élève jusqu'à 70 cm. Ramifié ou non, elle porte un épi floral assez typique, car au sommet aplati. Lorsqu'elles s'épanouissent, les fleurs tubulaires, rouges ou orangées, pendent de leur pédoncule allongé.

2.10.2 Besoins et mode de culture Aloe Maculata

Bien qu'il soit extrêmement résistant à la sécheresse, il est préférable de l'arroser régulièrement et abondamment durant sa période de croissance, soit de avril à fin septembre, afin qu'il se développe harmonieusement.

L'exposition doit être ensoleillée ou mi-ombre ; à l'intérieur, trouvez-lui une fenêtre ensoleillée. Plus il reçoit de soleil direct et plus il sera beau, trapu et coloré.

aloe maculata complètement au sec peut résister jusqu'à -11 °C .En hiver pluvieux, l'aloès zébré supporte très facilement le gel jusqu'à - 5 °C.

Entre -5 et -7 °C, ses feuilles sont abimées, mais il repart au printemps, pouvant même fleurir si le cœur ne gèle pas.

2.10.3 Entretien du Aloe Maculata

- Très facile d'entretien, l'Aloe demande très peu de soins.
- N'arrosez pas beaucoup et réduisez encore les apports en eau durant l'hiver.

2.10.4 Espèces et variétés de Aloe

Environ 270 espèces dans ce genre :

- Aloe vera, célèbre espèce médicinale
- Aloe haworthioides, petite espèce ressemblant à un Haworthia
- Aloe pilansii, Aloès en arbre géant
- Aloe arborescens, grande espèce arborescente
- Aloe plicatilis, magnifique espèce dichotomique
- Aloe ciliata, autre aloès buissonnant
- Aloe x spinosissima, bel hybride vigoureux
- Aloe rauhi, une espèce malgache
- Aloe variegata, aux feuilles en plume de perdrix
- Aloe polyphyla, espèce rare aux feuilles disposées en spirale

Conclusion

Nous avons décrit les plantes que nous avons utilisées dans notre application avec quelques conseils pour prendre soin d'elles comment les cultiver et de leurs besoins, c'est ce que donne le deuxième chapitre.

Chapitre 03

L'implémentation

Introduction

Dans ce chapitre de ruse, nous expliquerons en détail la façon dont nous travaillons et expliquerons tous les outils utilisés ainsi que les étapes de mise en œuvre

3.1 Langage de programmation utilisé (Python)

Python est un langage de programmation de haut niveau interprété (il n'y a pas d'étape de compilation) et orienté objet avec une sémantique dynamique. Il est très sollicité par une large communauté de développeurs et de programmeurs. Python est un langage simple, facile à apprendre et permet une bonne réduction du cout de la maintenance des codes.

Bibliothèques (packages) python encouragent la modularité et la réutilisabilité des codes. Python et ses bibliothèques sont disponibles (en source ou en binaires) sans charges pour la majorité des plateformes et peuvent être redistribués gratuitement. Python version 3.

3.2 L'environnement de développement utilisé (PyCharm (2021.3.3))

PyCharm est un environnement de développement intégré utilisé pour programmer en Python. Il permet l'analyse de code et contient un débogueur graphique. Il permet également la gestion des tests unitaires, l'intégration de logiciel de gestion de versions, et supporte le développement web avec Django. Écrit en : Java et Python est le Système d'exploitation : GNU/Linux, Microsoft Windows et macOS

3.3 Les Bibliothèques utilisées

3.3.1 Tensorflow

TensorFlow est un framework de programmation pour le calcul numérique qui a été rendu Open Source par Google en Novembre 2015. Depuis son release, TensorFlow n'a cessé de gagner en popularité, pour devenir très rapidement l'un des frameworks les plus utilisés pour le Deep Learning et donc les réseaux de neurones. Son nom est notamment inspiré du fait que les opérations courantes sur des réseaux de neurones sont principalement faites via des tables de données multi-dimensionnelles, appelées Tenseurs (Tensor). Un Tensor à deux dimensions est l'équivalent d'une matrice. Aujourd'hui, les principaux produits de Google sont basés sur TensorFlow : Gmail, Google Photos, Reconnaissance de voix .

3.3.2 Keras

Keras est une API de réseaux de neurones de haut niveau, écrite en Python et capable de fonctionner sur TensorFlow ou Theano. Il a été développé en mettant l'accent sur l'expérimentation rapide. Être capable d'aller de l'idée à un résultat avec le moins de délai possible est la clé pour faire de bonnes recherches. Il a été développé dans le cadre de l'effort de recherche du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System), et son principal auteur et mainteneur est François Chollet, un ingénieur Google. En 2017, l'équipe TensorFlow de Google a décidé de soutenir Keras dans la bibliothèque principale de TensorFlow. Chollet a expliqué que Keras a été conçue comme une interface plutôt que comme un cadre d'apprentissage end to end. Il présente un ensemble d'abstractions de niveau supérieur et plus intuitif qui facilitent la configuration des réseaux neuronaux indépendamment de la bibliothèque informatique de backend. Microsoft travaille également à ajouter un backend CNTK à Keras aussi.

3.3.3 NumPy

Numerical python est la bibliothèque la plus populaire de calcul scientifique en python .Ainsi ,elle permet d'effecture les calculs scientifique de base et de manipuler assez facilement les tableaux multidimensionnelles .en gros NumPy permet de faire :

- De l'analyse numérique .
- De l'algèbre linéaire .
- Du calcul matriciel .

3.3.4 OS

Une bibliothèque en langage python qui nous permet de remplir certaines commandes sur le système de déplacement qu'il s'agisse de Windows ou de Lunx peut être utilisée dans des

scripts et des programmes qui reposent sur l'interface graphique. en gros **OS** permet de faire :

- Connaissance des caractéristiques du système de déplacement
- Gestion des dossiers en termes de construction, de copie et de suppression
- Gestion des fichiers dans le système d'exploitation
- Traitement des variables

3.3.5 Tkinter

Le paquet tkinter (interface Tk») est l'interface Python standard de la boîte à outils Tk GUI. Tk et tkinter sont disponibles sur la plupart des plates formes Unix, ainsi que sur les systèmes Windows. utilise Tkinter pour créer rapidement des applications d'interface graphique car la bibliothèque contrôle aussi divers que des boutons et des zones de texte où il y a 15 éléments dans la bibliothèque

3.3.6Matplotlib

Matplotlib est une bibliothèque complète pour la création de visualisations statiques, animées et interactives en Python. Matplotlib rend les choses faciles et les choses difficiles possibles.

- Créez des graphiques de qualité de publication.
- Créez des figures interactives qui peuvent zoomer, faire un panoramique, mettre à jour.
- Personnalisez le style visuel et la mise en page.
- Exportez vers de nombreux formats de fichiers .
- Intégration dans JupyterLab et les interfaces utilisateur graphiques.
- Utilisez un large éventail de packages tiers basés sur Matplotlib.

3.3.7Random

Le module Python Random est un module intégré de Python qui est utilisé pour générer des nombres aléatoires. Ce sont des nombres pseudo-aléatoires. Ce module peut être utilisé pour :

- effectuer des actions aléatoires telles que la génération de nombres aléatoires.
- l'impression aléatoire d'une valeur pour une liste ou une chaîne.

3.3.8 Shutil

Shutil module est une abréviation pour les utilitaires shell, le module permet d'effectuer l'opération de fichier de haut niveau. Il peut fonctionner avec l'objet fichier et nous offre la possibilité de copier et de supprimer les fichiers. Il gère la sémantique de bas niveau telle que la création et la fermeture d'objets de fichier après avoir effectué toutes les opérations. Le module Python shutil est livré avec les nombreuses méthodes intégrées. Pour commencer à travailler avec ce module, nous devons d'abord l'importer dans notre fichier Python actuel.

3.3.9 PIL (Python Imaging Library)

est une bibliothèque de traitement d'image, La bibliothèque de fonctions peut être utilisée pour différents types d'activités, telles que:

- Création de miniatures, conversion d'images d'un format de fichier à un autre .
- Affichage d'images.
- Traitement d'images.

```
# Les bibliothèques python à importer
from os import makedirs
from os.path import isfile, join
from shutil import copyfile
from random import seed
from random import random
from os import listdir
import numpy as np
from PIL import Image
```

FIGURE 3.1 : Les Bibliothèques Python à Importes Dans La Page dataProcessing

```
import sys
from matplotlib import pyplot
from keras.utils import to_categorical
from keras.models import Sequential
from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Dense

from tensorflow.keras.layers import Flatten
#from keras.optimizers import SGD
from keras.optimizers import SGD
from keras.preprocessing.image import ImageDataGenerator
from keras.layers import Dropout
from keras.layers import BatchNormalization
import dataProcessing
```

FIGURE 3.2 : Les Bibliothèques Python à Importes Dans La Page creatModel

```
import numpy as np
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Flatten
from tensorflow.keras.optimizers import SGD
#from keras.optimizers import SGD
import dataProcessing
```

FIGURE 3.3 : Les Bibliothèques Python à Importes Dans La Page saveFinalModel

```
from tensorflow.keras.models import load_model
from tensorflow.keras.utils import to_categorical
import dataProcessing
```

FIGURE 3.4 : Les Bibliothèques Python à Importes Dans La Page evaluateFinalModel

```
import tkinter as tk
from tkinter import filedialog
from tkinter import *
from PIL import ImageTk, Image
import numpy
from tensorflow.keras.models import load_model
import shap
import dataProcessing

import tensorflow as tf
```

FIGURE 3.5 : Les Bibliothèques Python à Importes Dans La Page de l'interface

3.4 Data processing

- **Data set**

Dans cette étape, qui est l'étape de départ où nous avons recueilli des photos d'un groupe de plantes ornementales en prenant plusieurs photos de chaque plante de tous les côtés, où nous avons 504 photos que nous avons prises des pépinières situées au niveau de notre état.

Ensuite, l'organisation du dataset selon une structure appropriée toutes les images sont stockées dans le répertoire '**dataset**'. Chaque image est sauvegardée dans un fichier JPG son nom a la structure suivante:

nomDeLaPlante.numéro.jpg

Nous avons 10 variétés de plantes dans chaque plante, allant de 5 à 65 images par espèce.

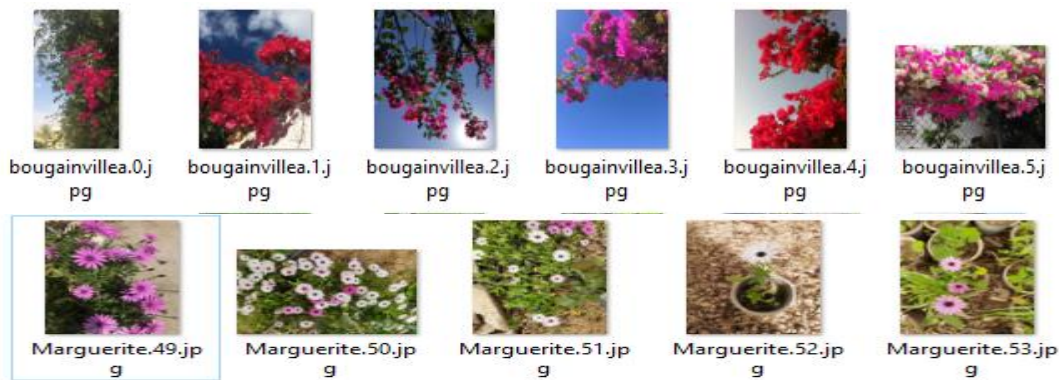


FIGURE 3.6 : Les images dans le dataset

- **Ensemble de données d'entraînement et de test (Training and testing dataset)**

La structure à créer est comme suit sous le répertoire dataset nous créons deux sous-répertoire 'train' et 'test' qui vont contenir eux-mêmes des sous-répertoires où leurs noms vont porter les noms des classes c'est à dire les noms des plantes.

Un sous-répertoire de classes contient tous les fichiers des images appartenant à cette classe.

- **Latechnique du Train-Test Split**
- **Entraînement des Données**

une partie des données utilisées pour construire le modèle d'entraînement . Plusieurs choix de conception peuvent être disponibles lors de la construction du modèle. Le réseau neuronal peut utiliser différents hyperparamètres pour le taux d'apprentissage ou pour la régularisation. Le même ensemble de données d'entraînement peut être essayé plusieurs fois sur différents choix pour les hyperparamètres ou des algorithmes complètement différents pour construire les modèles de plusieurs manières. Ce processus permet d'estimer la précision relative des différents paramètres de l'algorithme. Ce processus prépare le terrain pour la sélection du modèle, dans lequel le meilleur algorithme est sélectionné parmi ces différents modèles. Cependant, l'évaluation réelle de ces algorithmes pour sélectionner le meilleur modèle ne se fait pas sur les données d'entraînement, mais sur un ensemble de données de validation séparé pour éviter de favoriser les modèles sur adaptés.

Dans cette étape, nous avons fait une fonction que reformate les données d'entraînement et de test ou format qui sera utilisé par l'apprentissage automatique pour créer le modèle .

Nous l'avons nommé :

Load data (dataset_home)

Où nous avons déterminé la taille des images est 32 pixels en largeur et 32 pixels en hauteur et trois canaux de couleurs.

- **Teste des Données**

une partie des données utilisées pour tester la précision du modèle final (réglé). Il est important que les données de test ne soient même pas examinées pendant le processus de réglage des paramètres et de sélection du modèle pour éviter tout sur ajustement. Les données de test ne sont utilisées qu'une seule fois à la toute fin du processus. De plus, si l'analyste utilise les résultats des données de test pour ajuster le modèle d'une manière ou d'une autre, les résultats seront contaminés par les connaissances des données de test. L'idée que l'on ne soit autorisé à examiner un ensemble de données de test qu'une seule fois est une exigence extraordinairement stricte (et importante). Pourtant, il est fréquemment violé dans les repères de la vie réelle. La tentation d'utiliser ce que l'on a appris de l'évaluation finale de l'exactitude est tout simplement trop élevée.

L'approche Train-Test Split consiste à décomposer de manière aléatoire un ensemble de données. Une partie servira à l'entraînement du modèle de Machine Learning, l'autre partie permettra de le tester pour la validation.

En général, on réserve 70% à 80% des données du dataset pour l'entraînement. Les 20 à 30% restants seront exploités pour la Cross-Validation.

```
def create_structure(dataset_home):  
    # générateur de nombres aléatoires de départ  
    seed(1)  
    # définir le taux des images à utiliser pour la validation (test)  
    val_ratio = 0.25 # 25 %  
    # copier les images du jeu de données d'entraînement dans des sous-  
    répertoires  
    #l = listdir(dataset_home)
```

```

l = [f for f in listdir(dataset_home) if isfile(join(dataset_home, f))]
for file in l:
# fichier source
src = dataset_home + '/' + file
# Le sous-répertoire d'entraînement
dst_dir = '/train/'
if random() < val_ratio:
# le sous-répertoire de test (validation)
dst_dir = '/test/'
# le sous-répertoire de classe d'entraînement ou de test
classe = file.split('.')[0]
    makedirs(dataset_home + dst_dir + classe + '/', exist_ok=True)
    dst = dataset_home + dst_dir + classe + '/' + file
    copyfile(src, dst)

def load_data(dataset_home):
    Image_Width = 32
    Image_Height = 32
    Image_Size=(Image_Width,Image_Height)
    Image_Channels = 3
    train_class_directories = []
    for classe in listdir(dataset_home + '/train/'):
        train_class_directories = train_class_directories + [classe]
        num_train_samples = 0
    for classe in train_class_directories:
        num_train_samples = num_train_samples + len(listdir(dataset_home +
'/train/' + classe))
        x_train = np.empty((num_train_samples, Image_Width, Image_Height,
Image_Channels), dtype='uint8')
        y_train = np.empty((num_train_samples,), dtype='uint8')
        i = 0
    j = -1
    for classe in train_class_directories:
        j = j + 1
    for file in listdir(dataset_home + '/train/' + classe):
        image = Image.open(dataset_home + '/train/' + classe + '/' + file)
        image = image.resize(Image_Size)
        x_train[i] = np.asarray(image)
        y_train[i] = j
        i = i + 1
    test_class_directories = []
    for classe in listdir(dataset_home + '/test/'):
        test_class_directories = test_class_directories + [classe]
        num_test_samples = 0

    for classe in test_class_directories:
        num_test_samples = num_test_samples + len(listdir(dataset_home +
'/test/' + classe))
        x_test = np.empty((num_test_samples, Image_Width, Image_Height,
Image_Channels), dtype='uint8')
        y_test = np.empty((num_test_samples,), dtype='uint8')
        i = 0
    j = -1
    for classe in test_class_directories:
        j = j + 1
    for file in listdir(dataset_home + '/test/' + classe):
        image = Image.open(dataset_home + '/test/' + classe + '/' + file)
        image = image.resize(Image_Size)
        x_test[i] = np.asarray(image)
        y_test[i] = j
        i = i + 1

```

```
return (x_train, y_train), (x_test, y_test), len(listdir(dataset_home +  
'/train'))
```

FIGURE 3.7: Le Code de la page dataProcessing

3.5 Create Model

Sur cette page, nous avons ajouté des bibliothèques importants à notre travail, puis nous avons chargé l'ensemble de données (le dataset) Et catégorisez-le Par la fonction :

def load_dataset().

```
def load_dataset():
# charger le dataset
(trainX, trainY), (testX, testY), num_classes =
dataProcessing.load_data('dataset')
# la catégorisation des données
trainY = to_categorical(trainY)
testY = to_categorical(testY)
return trainX, trainY, testX, testY, num_classes

# mettre à l'échelle les pixels
def prep_pixels(train, test):
# convertir des nombres entiers en flottants
train_norm = train.astype('float32')
test_norm = test.astype('float32')
# normaliser à la plage 0-1
train_norm = train_norm / 255.0
test_norm = test_norm / 255.0
# renvoyer des images normalisées
return train_norm, test_norm
```

FIGURE 3.8 : Le Fonction De Chargé Le Data Set

Ensuite, nous avons défini le modèle CNN (réseaux de neurones convolutifs) on utilise la fonction :

def define_model(num_classes)

```
def define_model(num_classes):
model = Sequential()
model.add(
Conv2D(32, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same', input_shape=(32, 32, 3))
model.add(BatchNormalization())
model.add(Conv2D(32, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
model.add(Dropout(0.2))
model.add(Conv2D(64, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(64, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
model.add(Dropout(0.3))
model.add(Conv2D(128, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(BatchNormalization())
```

```

    model.add(Conv2D(128, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
    model.add(BatchNormalization())
    model.add(MaxPooling2D((2, 2)))
    model.add(Dropout(0.4))
    model.add(Flatten())
    model.add(Dense(128, activation='relu',
kernel_initializer='he_uniform'))
    model.add(BatchNormalization())
    model.add(Dropout(0.5))
# num_classes est le nombre de classes des plantes
model.add(Dense(num_classes, activation='softmax'))
# compiler le modèle avec l'optimisateur SGD
opt = SGD(lr=0.001, momentum=0.9)
    model.compile(optimizer=opt, loss='categorical_crossentropy',
metrics=['accuracy'])
return model

```

FIGURE 3.9 : Le Fonction qui Dèfini Le Model CNN

Après, nous avons fait une fonction pour tracer les courbes de diagnostic d'apprentissage et exécuter le faisceau de test pour évaluer un modèle.

Def summarize_diagnostics (history)

Def num_test_harness()

```

def summarize_diagnostics(history):
    pyplot.subplot(211)
    pyplot.title('Perte d\'entropie croisée et Précision de la
classification')
    pyplot.ylabel('Perte d\'entropie croisée')
    pyplot.plot(history.history['loss'], color='blue', label='train')
    pyplot.plot(history.history['val_loss'], color='orange', label='test')
    pyplot.subplot(212)
    pyplot.xlabel('Nombre d\'itérations')
    pyplot.ylabel('Précision de la classification')
    pyplot.plot(history.history['accuracy'], color='blue', label='train')
    pyplot.plot(history.history['val_accuracy'], color='orange',
label='test')
    filename = sys.argv[0].split('/')[0]
    pyplot.savefig(filename + '_plot.png')
    pyplot.close()
# exécuter le faisceau de test pour évaluer un modèle
def run_test_harness():
    trainX, trainY, testX, testY, num_classes = load_dataset()
    trainX, testX = prep_pixels(trainX, testX)
    model = define_model(num_classes)
    datagen = ImageDataGenerator(width_shift_range=0.1,
height_shift_range=0.1, horizontal_flip=True)
    it_train = datagen.flow(trainX, trainY)
    steps = int(trainX.shape[0]/10)
    history = model.fit(it_train, validation_data=(testX, testY),
epochs=400, batch_size=10, verbose=1)
    _, acc = model.evaluate(testX, testY, verbose=0)
    print('La précision est > %.3f' % (acc * 100.0))
    summarize_diagnostics(history)

```

FIGURE 3.10 – Le fonction Pour Trace Les Courbes est évaluer Un Modèle

3.6 Save Final Model

Durant le processus de création de modèle, un modèle se trouve dans la mémoire et est accessible tout au long du cycle de vie de l'application. Toutefois, dès la fin de l'exécution de l'application, le modèle n'est plus accessible s'il n'est pas enregistré localement ou à distance. En général, les modèles sont utilisés après l'entraînement dans d'autres applications à des fins d'inférence ou de réentraînement. C'est pourquoi il est important de stocker le modèle.

C'est pourquoi nous avons utilisé :

model.save(filepath)

pour enregistrer un Keras modèle en une seule fichier

```
def load_dataset():
# charger le dataset
(trainX, trainY), (testX, testY), num_classes =
dataProcessing.load_data('dataset')
# encode le valeurs de la cible
trainY = to_categorical(trainY)
testY = to_categorical(testY)
return trainX, trainY, testX, testY, num_classes

# mettre à l'échelle les pixels
def prep_pixels(train, test):
# convertir des nombres entiers en flottants
train_norm = train.astype('float32')
test_norm = test.astype('float32')
# normaliser à la plage 0-1
train_norm = train_norm / 255.0
test_norm = test_norm / 255.0
# renvoyer des images normalisées
return train_norm, test_norm

# définir le modèle cnn
def define_model(num_classes):
model = Sequential()
model.add(
Conv2D(32, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same', input_shape=(32, 32, 3))
model.add(Conv2D(32, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(Conv2D(64, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(128, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(Conv2D(128, (3, 3), activation='relu',
kernel_initializer='he_uniform', padding='same'))
model.add(MaxPooling2D((2, 2)))
model.add(Flatten())
model.add(Dense(128, activation='relu',
kernel_initializer='he_uniform'))
model.add(Dense(num_classes, activation='softmax'))
# compile model
```

```
opt = SGD(lr=0.001, momentum=0.9)
    model.compile(optimizer=opt, loss='categorical_crossentropy',
metrics=['accuracy'])
return model

# exécuter le faisceau de test pour évaluer un modèle
def run_test_harness():
# charger le dataset
trainX, trainY, testX, testY, num_classes = load_dataset()
# préparer les données de pixels
trainX, testX = prep_pixels(trainX, testX)
# définir le modèle
model = define_model(num_classes)
# fit le modèle
Xtrain = np.append(trainX, testX, axis=0)
    Ytrain = np.append(trainY, testY, axis=0)
    model.fit(Xtrain, Ytrain, epochs=100, verbose=0)
# sauvegarder le modèle
model.save('final_model.h5')
```

FIGURE 3.11 : Le Code De La Page Save Final Model

3.7 Evaluate Final Model

L'évaluation de modèle est le processus d'utilisation de différentes mesures d'évaluation pour comprendre les performances d'un modèle d'apprentissage automatique, ainsi que ses forces et ses faiblesses. L'évaluation du modèle est importante pour évaluer l'efficacité d'un modèle au cours des phases de recherche initiales, et elle joue également un rôle dans la surveillance du modèle.

```
from tensorflow.keras.models import load_model
from tensorflow.keras.utils import to_categorical
import dataProcessing

def load_dataset():
    # Charger le dataset
    (_, _), (testX, testY), _ = dataProcessing.load_data('dataset')
    # Valeurs cibles d'encodage
    testY = to_categorical(testY)
    return testX, testY

# Mettre à l'échelle les pixels
def prep_pixels(test):
    # Convertir des nombres entiers en flottants
    test_norm = test.astype('float32')
    # Normaliser à la plage 0-1
    test_norm = test_norm / 255.0
    # Renvoyer des images normalisées
    return test_norm

# Exécuter le faisceau de test pour évaluer un modèle
def run_test_harness():
    # Charger le dataset
    testX, testY = load_dataset()
    # Préparer les données de pixels
    testX = prep_pixels(testX)
    # Charger le modèle
    model = load_model('final_model.h5')
    # évaluer le modèle sur l'ensemble de données de test
    _, acc = model.evaluate(testX, testY, verbose=0)
    print('La précision est > %.3f' % (acc * 100.0))
```

FIGURE 3.12 : Le Code De La Page Evaluate Final Mode

3. 8. L'interface graphique

Pour l'interface, nous avons choisi d'avoir un design simple et convivial qui a trois boutons :

- Premier bouton pour le processus de téléchargement de l'image que l'utilisateur voit reconnaître .



FIGURE 3.13 : l' image de l'interface

- Le deuxième bouton pour le processus de classification, qui est pressé dès que l'image est téléchargée pour trouver le nom de cette plante avec une petite définition à côté de l'image .

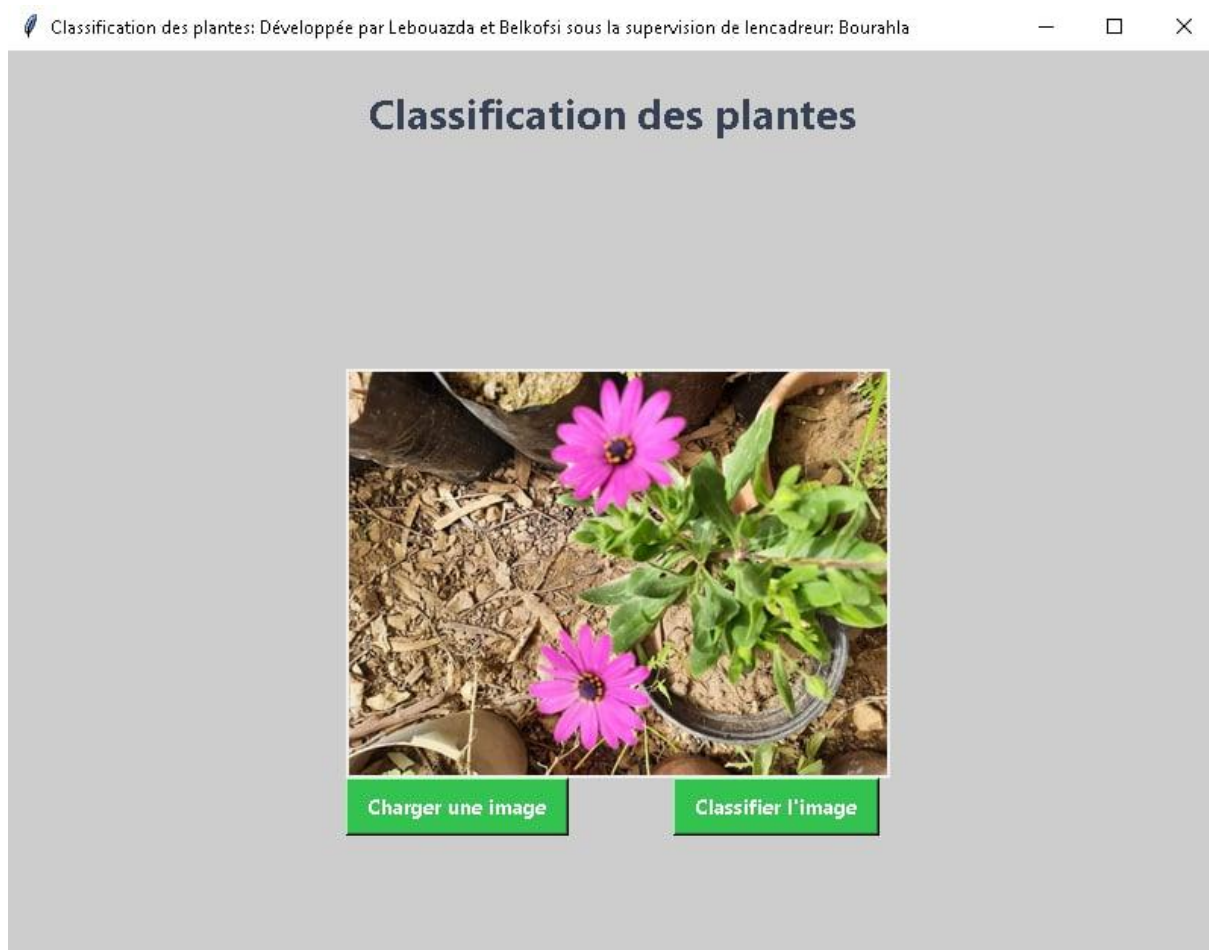


FIGURE 3.14– l'image de l'interface quand on appuie charger une image

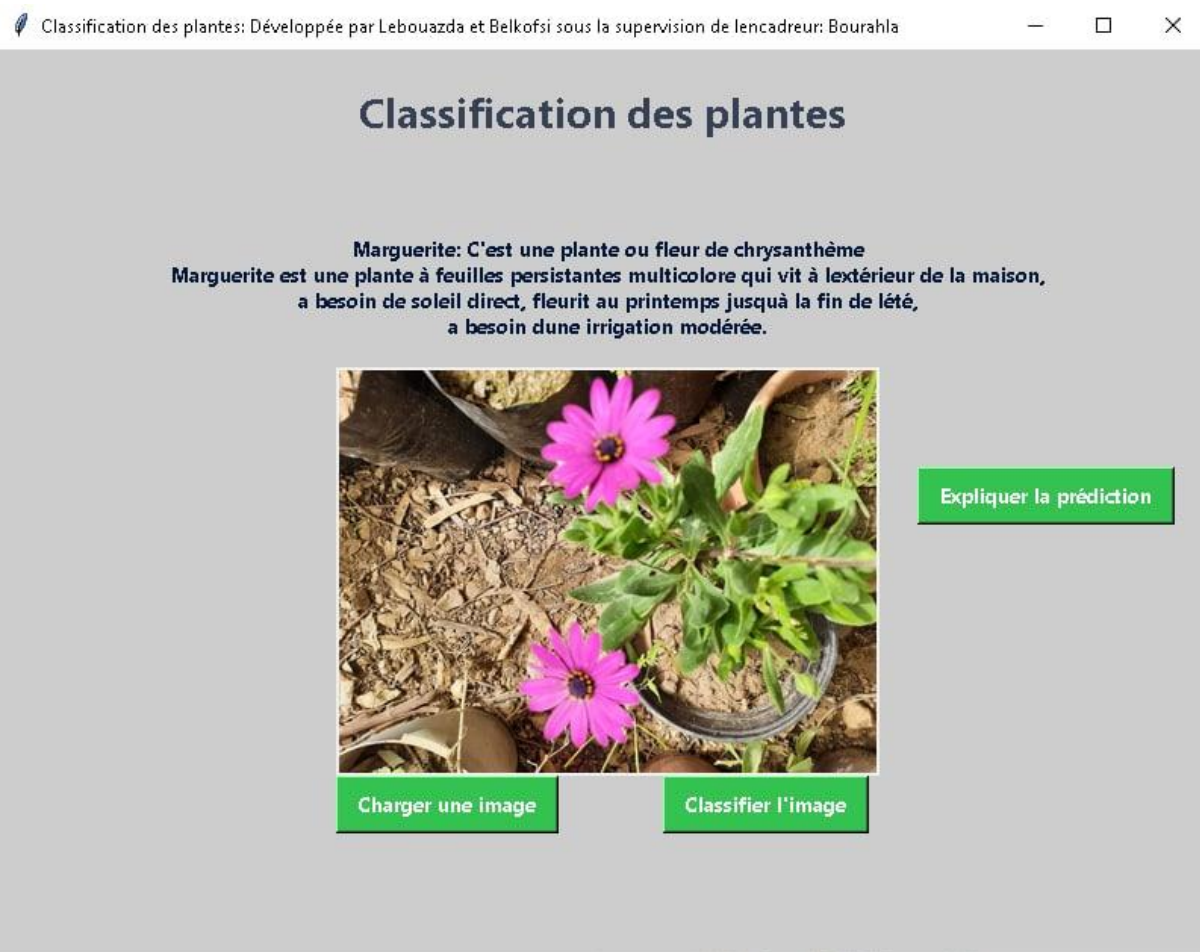


FIGURE 3.15 : l'image de l'interface quand on appuie classifier l'image

- Le troisième bouton pour avoir des explications

Figure 1

- □ ×

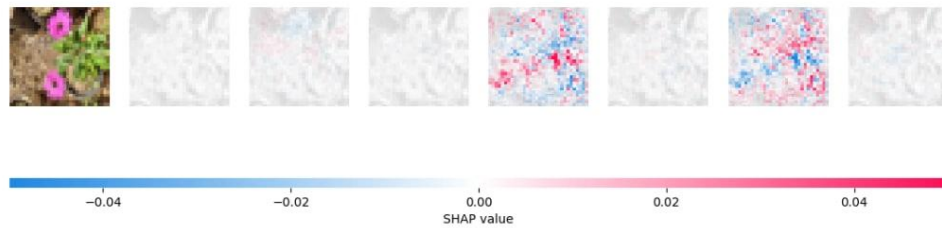


FIGURE 3.16 : l'image de l'interface quand on appuie explications

```
# Sélectionner un ensemble d'exemples d'arrière-plan pour répondre à une
prédiction
(x_train, _), (x_test, _) = dataProcessing.load_data('dataset')
x = numpy.append(x_train, x_test, axis=0)
background = x[numpy.random.choice(x.shape[0], len(x), replace=False)]
# Chargement du modèle
model = load_model('final_model.h5')
# Définir l'explication profonde
e = shap.DeepExplainer(model, background)
# Dictionnaire pour étiqueter toutes les classes des plantes.
classes = {
1: 'Bougainvillea: C\'est une plante',
2: 'Dianthus: C\'est une plante',
3: 'Gazania: C\'est une plante',
4: 'Géranium: C\'est une plante',
5: 'Marguerite: C\'est une plante',
6: 'Petunia: C\'est une plante',
7: 'Rosier de Chine: C\'est une plante',
8: 'Calendula Officianalis: C\'est une plante',
9: 'Aloe Maculata: C\'est une plante',
}
}
# initialiser GUI
top = tk.Tk()
top.geometry('800x600')
top.title('Classification des plantes: Développée par Lebouazda et Belkofsi
sous la supervision de l'encadreur: Bourahla')
top.configure(background='#CDCDCD')
label = Label(top, background='#CDCDCD', font=('arial', 15, 'bold'))
sign_image = Label(top)

# La fonction de classification
def classify(file_path):
```

```

global label_packed
    image = Image.open(file_path)
    image = image.resize((32, 32))
    image = numpy.expand_dims(image, axis=0)
    image = numpy.array(image)
    image = image / 255
#pred = sum(model.predict_classes([image]))
pred = sum(model.predict([image]))
pclasses = numpy.argmax(pred)
sign = classes[pclasses]
label.configure(background='#011638', text=sign)
show_explain_button(file_path)

# Fonction d'explication de prédiction
def explain(file_path):
    image = Image.open(file_path)
    image = image.resize((32, 32))
    image = numpy.expand_dims(image, axis=0)
    image = numpy.array(image)
    image = image / 255
shap_values = e.shap_values(image)

shap.image_plot(shap_values, image)

# Afficher le bouton de classification
def show_classify_button(file_path):
    classify_b = Button(top, text="Classifier l'image", command=lambda:
classify(file_path), padx=10, pady=5)
    classify_b.configure(background='#364156', foreground='white',
font=('arial', 10, 'bold'))
    classify_b.place(relx=0.76, rely=0.36)

# Afficher le bouton d'explication
def show_explain_button(file_path):
    classify_b = Button(top, text="Expliquer la prédiction",
command=lambda: explain(file_path), padx=10, pady=5)
    classify_b.configure(background='#364156', foreground='white',
font=('arial', 10, 'bold'))
    classify_b.place(relx=0.76, rely=0.46)

# Chargement de l'image à classifier
def upload_image():
    try:
        file_path = filedialog.askopenfilename()
        uploaded = Image.open(file_path)
        uploaded.thumbnail(((top.wininfo_width() / 2.25), (top.wininfo_height()
/ 2.25)))
        im = ImageTk.PhotoImage(uploaded)
        sign_image.configure(image=im)
        sign_image.image = im
        label.configure(text='')
        show_classify_button(file_path)
    except:
        pass

upload = Button(top, text="Charger une image", command=upload_image,
padx=10, pady=5)
upload.configure(background='#364156', foreground='white', font=('arial',
10, 'bold'))
upload.pack(side=BOTTOM, pady=50)
sign_image.pack(side=BOTTOM, expand=True)

```

```
label.pack(side=BOTTOM, expand=True)
heading = Label(top, text='Classification des plantes', pady=20,
font=('arial', 20, 'bold'))
heading.configure(background='#CDCDCD', foreground='#364156')
heading.pack()
top.mainloop()
```

FIGURE 3.17– Le code de la page GUI

- Le figure suivante (**FIGURE 3.18**)montre la perte d'entropie et la précision de la classification de motre modèle CNN la précision a 98.02%

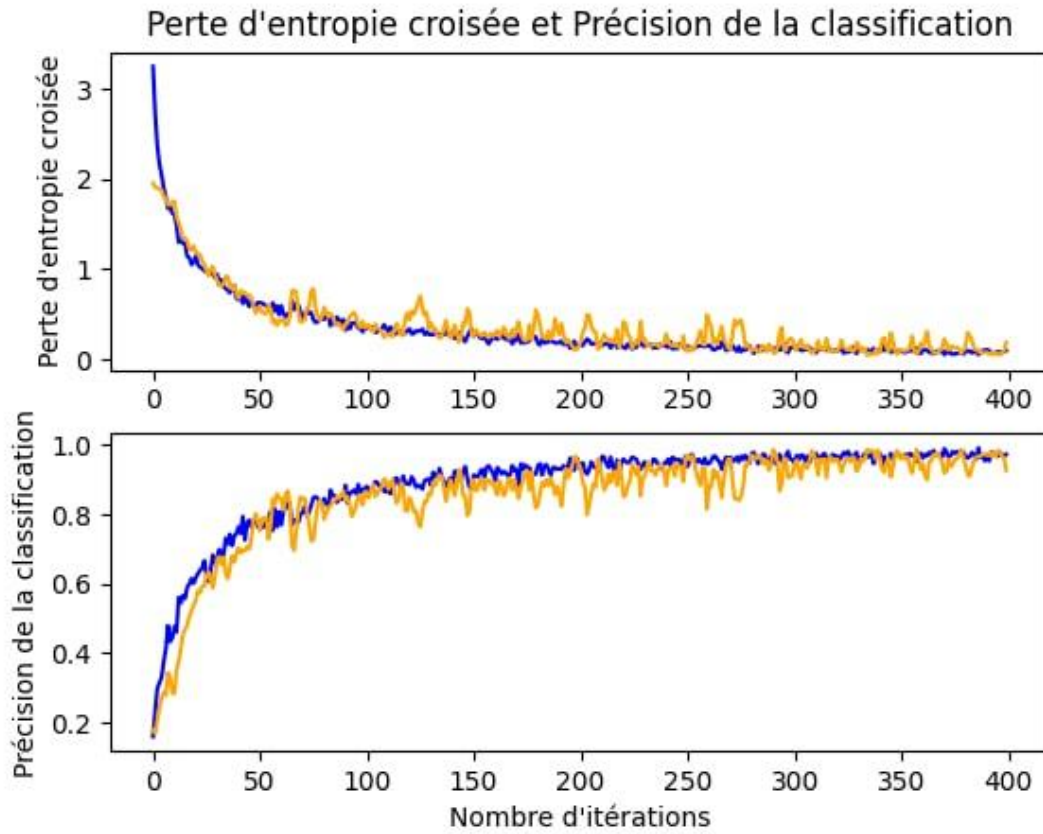


FIGURE 3.18– le perde d'entropie et la précision

3.9. Main

La fonction main est une fonction principale de Python est le début de tout programme Python.

Ce script est pour exécuter tous les autres modules (de création de la structure de données a l'evaluation de model final)apes l'exécution de ce script cela ne montre un nouvelle model qui s'appel 'model.h5', model h5 est un format de fichier pour stocker des données structurées, ce n'est pas un modèle en soi. Keras enregistre les modèles dans ce format car il peut facilement stocker les poids et la configuration du modèle dans un seul fichier.

```
import createModel
import dataProcessing
import evaluateFinalModel
import saveFinalModel

# Création de la structure de données
dataProcessing.create_structure('dataset')
# Création et test du modèle
createModel.run_test_harness()
# Sauvegarde du modèle final
saveFinalModel.run_test_harness()
# Evaluation du modèle final
evaluateFinalModel.run_test_harness()
```

FIGURE 3.19 : Le code de la page MAIN

Conclusion

Dans ce chapitre, nous avons parlé des différents langages de programmations, des librairies décrit. Nous avons également décrit les différentes phases nécessaire à la mise en place de la solution

Les données sont les clés pour avoir un système performant et efficace.

Conclusion et Perspective

La plante est l'élément essentiel de la vie sur Terre, car elle constitue 80% de la nourriture que nous mangeons et 98% de l'oxygène que nous respirons, et 80% de la population mondiale dépend des plantes avec des soins de santé. nécessaire de connaître le type de plante et d'en prendre soin, à travers lesquelles l'identité de la plante a été identifiée sans connexion Internet.

Bien qu'il existe plusieurs méthodes de détection automatisée des plantes, les recherches continuent pour augmenter leur efficacité et précision. Dans ce travail, plusieurs méthodes d'apprentissage profond ont été explorées afin de classer et d'identifier automatiquement les plantes à partir d'images de feuilles. Les modèles développés ont pu identifier les plants choisies 10 plantes. Nous avons collecté les images sur Pépinières pour former notre ensemble de données. Afin d'étendre l'ensemble de données, la stratégie de fine-tuning a atteint une précision élevée de 98.02% sur l'ensemble d'entraînement, 97,5% sur celui de validation et une précision de 99.99% sur l'ensemble de test.

Des perspectives intéressantes sont envisagées. On pourrait, par exemple :

1. Collecter d'autres images de ces 10 types de plants dans le but de rendre notre modèle beaucoup plus fiable et précis .
2. Élargir le champs d'identification du système à d'autres plants .
3. proposer un système de segmentation d'images dans l'optique de permettre la détection .

BIBLIOGRAPHIE

Site internet:

- [1] .Botanique Bougainvillea
<https://www.aujardin.info/plantes/bougainvillier>
- [2] Espèces et variétés de Bougainvillea
<https://www.aujardin.info/plantes/bougainvillier>
- [3] Botanique Marguerit
<https://www.aujardin.info/plantes/marguerit>
- [4] Espèces et variétés de Marguerite
<https://www.aujardin.info/plantes/marguerite>
- [5] .Botanique Gèraniume zolana
<https://www.aujardin.info/plantes/gèraniumezonala>
- [6] Espèces et variétés de Gèraniume zolana
<https://www.aujardin.info/plantes/gèraniumezolana>
- [7] .Botanique Pètunia
<https://www.aujardin.info/plantes/pètunia>
- [8] .Espèces et variétés de Pètunia
<https://www.aujardin.info/plantes/pètunia>
- [9] .Botanique Dianthus
<https://www.aujardin.info/plantes/dianthus>
- [10] .Espèces et variétés de Dianthus
<https://www.aujardin.info/plantes/dianthus>
- [11] .Botanique Gazania
<https://www.aujardin.info/plantes/gazania>
- [12] Espèces et variétés de Gazania
<https://www.aujardin.info/plantes/gazania>
- [13] Botanique Aptenia cordifolia
<https://www.aujardin.info/plantes/apteniacordifolia>
- [14] Espèces et variétés de Aptenia cordifolia
<https://www.aujardin.info/plantes/apteniacordifolia>
- [15] Botanique Rosier de chine
<https://www.aujardin.info/plantes/rosierdechine>
- [16] Espèces et variétés de Rosier de chine

<https://www.aujardin.info/plantes/rosierdechine>

[17] Botanique Calendula Officinalis

<https://www.aujardin.info/plantes/calendulaofficinalis>

[18] Espèces et variétés de Calendula Officinalis

<https://www.aujardin.info/plantescalendulaofficinalis>

[19] Botanique Aloe Maculata

<https://www.aujardin.info/aloemaculata>

[20] Espèces et variétés Aloe Maculata

<https://www.aujardin.info/aloemaculata>

الملخص

في هذه المذكرة تناولنا مجموعة من المفاهيم وناقشناها كالذكاء الاصطناعي والتعرف على النباتات وأهميتها الكبيرة الى أن توصلنا الى طرح فكرتنا الخاصة الا وهي منصة للتعرف على النباتات وفي هذا الصياغ طرحنا طريقة انشائنا للفكرة من الأدوات والصور وصولا الى ما أنجزناه وقدمنا

الكلمات المفتاحية: (الذكاء الاصطناعي، النباتات، تصنيف، الشبكات العصبية)

Abstract

In this note, we have addressed a range of concepts and discussed them such as artificial intelligence, plant recognition and their great importance.

Until we found our own idea which is a platform to get to know plants and in this formulation, we put forward the way we created the idea of tools and images until we accomplished and presented .

Key words : (artificial intelligence ,plants ,classification ,cnn)

RÉSUMÉ

Dans cette note, nous avons abordé une gamme de concepts et les avons discutés tels que l'intelligence artificielle, la reconnaissance des plantes et leur grande importance.

Jusqu'à ce que nous trouvions notre propre idée qui est une plate-forme pour apprendre à connaître les plantes et dans cette formulation, nous avons mis en avant la façon dont nous avons créé l'idée d'outils et d'images jusqu'à ce que nous ayons accompli et présenté.

Les mots clés : (l'intelligence artificielle ,plantes ,classification ;cnn)