

UNIVERSITÉ NATIONALE DU VIETNAM À HANOÏ
INSTITUT FRANCOPHONE INTERNATIONAL



Option : Systèmes Intelligents et Multimédia (SIM)

Promotion : XXII

**TPE : DEEP LEARNING POUR LA CLASSIFICATION
D'IMAGE BIOLOGIQUE**

Rapport de la partie théorique

Etudiant :

DAOUDA KADRI Saidi

Encadreur :

Dr. HO TUONG VINH

Année académique 2017-2018

Table des matières

1	Introduction	4
2	Domaine d'étude	4
2.1	Problème vise	4
2.2	Problème à résoudre	4
2.3	Difficultés Possibles	4
2.4	Travaux théorique	5
2.5	Travaux pratique	5
2.6	Outils, Technique, Algorithme	5
2.7	Résultats attendus	5
3	Définition des termes techniques	5
4	ÉTUDE DE L'EXISTANT	6
4.1	K plus proches voisins(KNN)	6
4.2	Machine à vecteur de support (SVM)	6
4.3	les arbres de décisions	7
4.4	Réseau de neurones artificielle	7
4.5	Réseaux de neurones convolutifs(CNN)	7
4.6	Étude comparative	13
5	Solution proposé	14
5.1	Contexte général	14
5.2	Problématique	14
5.3	Description de la base de donnée Breakhis	14
5.4	Solution proposée	15
5.5	Contribution	15
5.6	Présentation des outils	16
5.6.1	Le software	16
5.6.2	Theamo	16
5.6.3	Keros	17
5.6.4	Autres frameworks	17
5.7	Outils à utiliser	18
5.7.1	Hardware	18
5.7.2	Software	18
6	Planification des tâches	18
7	Expérimentation	19
7.1	Nation de base	19

8	Architecture	21
9	Initialisation des paramètres	22
9.1	Glorot initialization	22
10	Régularisation	22
10.1	Dropout	22
11	Résultat de l'apprentissage	23
12	Conclusion et perspectives	25

Table des figures

1	les performances de type1 et type2	6
2	CNN confusion matrix classes A,B and C	8
3	les performances de type1 et type2	9
4	les taux de précision de type1, type2 et type3	9
5	Résultat de la 1ère méthode	11
6	Résultat de la 2ème méthode	12
7	Comparaison de méthode	13
8	Tumeur Bénigne	15
9	Tumeur Maligne	16
10	Architecture du Prototype à développé	17
11	zero padding	19
12	Non zero padding	20
13	zero padding	21
14	Dropout	23
15	Taux apprentissage	24

Liste des tableaux

1	Architecture	8
2	Architecture	10
3	Planification tâche	18

1 Introduction

Dans le monde actuel,nous voyons des milliers d'images qu'on crois identique alors qu'ils existent des différences qui nous échapper.Pour pallier à ce problème de grand envergure que nous utiliserons la classification qui consiste à identifier les images à partir de leurs caractéristiques (couleur, forme,contour,etc...) en fin de leur catégorisé.

l'apprentissage automatique est la capacité qu'un application a pouvoir prendre une décision à partir des données ;Dans le contexte de la biologie anatomie la classification permet de détecter les maladies des patients à fin d'aider les médecins dans la prise de décisions.

Dans ce travail l'attente est de pouvoir améliorer système de classification qui permet de prendre a l'entrée l'image donnée par l'utilisateur, de vérifier ces contenus en fin de déterminer sa catégorie dans la base image. Il existe plusieurs solutions pour classifier des images , mais pour notre cas nous utiliserons le deep learning.

2 Domaine d'étude

Après analyse approfondie de notre projet,Nous avons constate que le champs d'étude du sujet fait appel au domaine de la science bio-informatique,le traitement automatique des données.De plus il fait appel au traitement d'image et la vision par ordinateur en basent sur le domaine de l'apprentissage profond en fin de pouvoir proposer une solution qui peut s'adapter aux image biologique

2.1 Problème vise

la biologie est une science qui étudie l'ensemble des êtres vivants.Dans le cas de notre projet nous étudierons la biologie anatomie est qui est une science de la médecine qui étudie l'organisation structurelle des être vivants y compris les organes et les tissus qui les constituent.concrètement notre problème est de pouvoir détecte les maladies. Il nous est demande de faire un état d'arts des travaux dans le domaine de la classification d'image biologique

2.2 Problème à résoudre

Améliorer un système de classification d'image en se basant sur des failles et limites des systèmes existant.

2.3 Difficultés Possibles

Notre travail sera, comme tout sujet scientifique.rencontrera des difficultés de divers ordres. Parmi celles-ci nous pouvons citer les problèmes lies :

- • Au manques de connaissances dans le domaine ;
- • A l'accès a la documentation ;
- • Non maîtrise du langage de programmation python ;

- • choix du domaine d'implémentation

2.4 Travaux théorique

le but du travaux théorique est de faire des recherches comme toute projet scientifique,étudier les articles et les documents en référence avec le thème a fin de faire une synthèse des avantages et inconvénients sur des méthodes et techniques utilisent. C'est a partir de cette étude que nous pouvons décider si on doit proposé une nouveaux solution ou améliorer une solution existant

2.5 Travaux pratique

Après la choix de la solution,l'option suivant est la modélisation de l'application en se basant sur les réseaux de neurone du deep learning, la grand partie de cette section est d'effectuer une comparaison pour vérifier si le prototype développe répond au objectifs énumères dans l'état de l'art.

2.6 Outils,Technique,Algorithme

Notre projet de TPE consiste à développer un prototype,pour ce là nous utiliserons l'algorithme de deep learning qui nous permet d'effectuer une extraction de caractéristique automatique sur les images à de les classes par catégorie.

2.7 Résultats attendus

la fin de tout recherche scientifique est suivi de résultats,pour ce travail les résultats à attendre sont :

- • Avoir une connaissance solide dans le domaine de deep learning ;
- • Avoir une synthèse sur des avantages et inconvénients par rapport aux méthodes existant ;
- • Pouvoir améliorer au proposer autre méthode ;
- • Avoir une base de donnée solide pour l'entraînement du prototype ;

3 Définition des termes techniques

- **Intelligence artificielle** :est le domaine de recherche visant a reproduire des systèmes artificiels, les diverses capacités cognitives de l'être humaine.
- **Réseau de neurone** :les réseaux de neurones artificiels sont des réseaux fortement connectes de processeurs élémentaire fonctionnant en parallèle. chaque processeur élémentaire calcule une sortie unique sur la base des information qu'il reçoit[1].
- **Machine learning** :c'est un type d'intelligence artificielle qui confère au ordinateur la capacité d'apprendre sans être explicitement programme.
- **Deep learning** :est un sous ensemble de méthode d'apprentissage automatique fondée sur l'apprentissage de modèle de données.

4 ÉTUDE DE L'EXISTANT

Dans cette partie nous allons faire l'étude sur les articles qui parle de notre problème afin d'avoir une vue générale sur les méthodes utilisées, l'architecture des réseaux de neurones.

Carlos affonso et al dans[2] ont utilisé d'une part les algorithmes de machine learning(KNN,SVM,AD,RNA) associe a un descripteur,d'autre part le réseau de neurone convolutionnel a fin de faire la classification de trois type de bois(A,BetC) d'une société brésilienne les deux méthodes.

L'ensemble des données est de 374 images dont 144 de type A, 177 de type B et 53 de type C avec une résolution de 96*27 pixels.

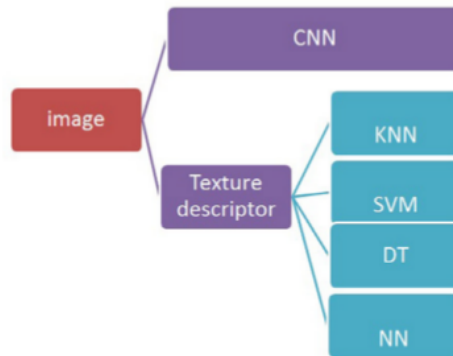


FIGURE 1 – les performances de type1 et type2

4.1 K plus proches voisins(KNN)

L'algorithme K plus proche voisin est l'un des algorithmes les plus simple en apprentissage pour la classification. L'idée est la suivante, après l'extraction des caractéristiques ils ont appliqué le KNN pour classier les trois type de bois, ils ont eu comme taux de précision 82,11% et $k=0,703$ après avoir regroupe les classes en deux A et BC le taux est de 92,50% et $k=0,8456$.

Avantages :

- simple à concevoir

Inconvénient

- Sensible aux bruits
- Pour un nombre de variable prédictives très grands, le calcul de la distance devient très coûteux.

4.2 Machine à vecteur de support (SVM)

Machine à vecteur de support sont des algorithmes de classification qui utilise la méthode linéaire et non-linéaire (si le nombre de classe est supérieur à deux). La méthode non-linéaire sur les trois classes a permis d'obtenir un taux de 80,73 et $k=0,660$. Après avoir regroupe les trois classes en deux (A et BC) nous avons constaté une augmentation au niveau de leur taux de précision 92,49% et $k=0,8544$. **Avantages :**

- C'est un modèle simple à comprendre et à interpréter

Inconvénient :

- Sensible aux bruits
- Pour un nombre de variable prédictives très grands, le calcul de la distance devient très coûteux.

4.3 les arbres de décisions

les algorithmes d'arbre de décisions sont aussi utilisés pour la classification, les entrées et les sorties peuvent être discrètes ou continues. Cette algorithmique leur a permis d'avoir comme taux 81,28% et $k=0,688$ sur l'ensemble des données. Après avoir réorganisé les classes en A, BC on voit bien que leur taux a considérablement changé 93,31% et $k=0,8598$. **Avantages :**

- ils sont faciles à comprendre
- Ils permettent de sélectionner l'option la plus appropriée parmi plusieurs

Inconvénient :

- L'existence d'un risque de sur-apprentissage si l'arbre devient très complexe.

4.4 Réseau de neurones artificielle

Perceptron multicouche est un classifieur de type réseaux de neurones. L'avantage de cette algorithmique il est capable de compresser l'ensemble des informations de son apprentissage afin d'éviter de toujours consulter le modèle d'entraînement lorsque nous voulons faire de nouvelles prédictions.

L'application de cette algorithmique de perceptron a permis d'atteindre un taux de 81,26% et $k=0,6807$ pour les trois types de bois et un taux de 93,56% avec $k=0,8673$ pour les classes A et BC.

Avantages :

- Accepte des données bruitées et la classification non-linéaire
- Architecture simple

Inconvénient :

- le nombre de couches cachées et les neurones dans les couches est indéfini

4.5 Réseaux de neurones convolutifs(CNN)

Les réseaux de neurones convolutifs sont des réseaux à plusieurs couches, ils sont spécialisés dans les tâches de reconnaissance, classification et prédiction. Les réseaux de neurones convolutifs ont la capacité d'extraire automatiquement les caractéristiques des images. Ils ont utilisé le CNN avec deux topologies différentes le 20-10-10-4-4 et le 25-10-10-4-4 pour réaliser leur expérience. Après avoir entraîné 10 fois de suite le réseau. Les tests leur ont permis d'obtenir le taux de performance 77,69% avec $k=0,6115$ pour les trois classes et 79,07% avec $k=0,5493$ pour les classes A, BC.

La figure 2 nous montre que l'algorithme a effectué des confusions lors de la classification de certaines images.

real \ predicted	A	B	C
A	117	27	0
B	17	158	2
C	15	28	10

FIGURE 2 – CNN confusion matrix classes A,B and C

Toto Haryanto et al dans[3] ont utilisée comment donnée les images en couleur de taille 775 * 522 pixels avec une résolution de 0,68Hm/pixels.L'ensemble des données utilisée pour l'apprentissage est insignifiant 37 images bénigne et 48 malignes.Pour résoudre ce problème ils ont décidé de génère des sous-échantillonnages d'image de taille 50 * 50 à partir d'image d'entrée. cette méthode leur a permis d'avoir une total de 515 images.

En générale l'architecture des réseau de neurone convolution est compose de trois partis :l'entrée d'image,l'extraction des caractéristiques et la classification.Dans leur projet ils ont utilisé trois combinaisons de couches à partir de six(type1),huit(type2) et dix(type3). A chaque entrée d'image, vingt images sont sélection par le processus de propagation pour la mise a jour du poids de réseau.

TABLE 1 – Architecture

Nome	layer	conv	maxpool	dense	output
Type1	6	2	2	1	1
type2	8	3	3	1	1
type3	10	4	4	1	1

Dans type1 l'expérience est réalisé avec des données réelles sans pré-traitement et avec pré-traitement,Environs six heures de temps pour chaque séries d'images.les résultats expérimentaux des images sans pré-traitement sont : 73,91% pour bénigne,91,67% pour malignes et 82,98% de performance globale. Pour des images avec pré-traitement on a une performance globale de 68,63%, pas mieux que les images réelles.

Dans type2 on constate que l'ajout de couches à vraiment impacte le temps d'apprentissage. l'architecture type2 nécessite sept heures pour chaque ensemble d'image, le même teste effectué sur type1 est applique a type2.Les résultats montrent un taux de performance globale 81,91% pour les images sans pré-traitement et 74,51% avec pré-traitement à partir de ces résultats nous pouvons dire que les images pré-traiter ne permettent pas d'avoir les meilleurs performances.

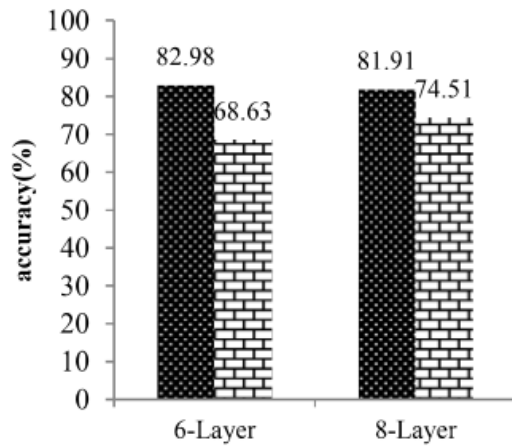


FIGURE 3 – les performances de type1 et type2

Par conséquent ils ont appliqué uniquement les images sans pré-traitement dans l'architecture type3 pour l'apprentissage, l'algorithme avait mis huit heures de temps pour chaque catégorie d'images. Avec la même base de données on constate les résultats suivantes : 81,62% pour bénignes, 95,83% pour malignes et une performance globale de 89,36%.

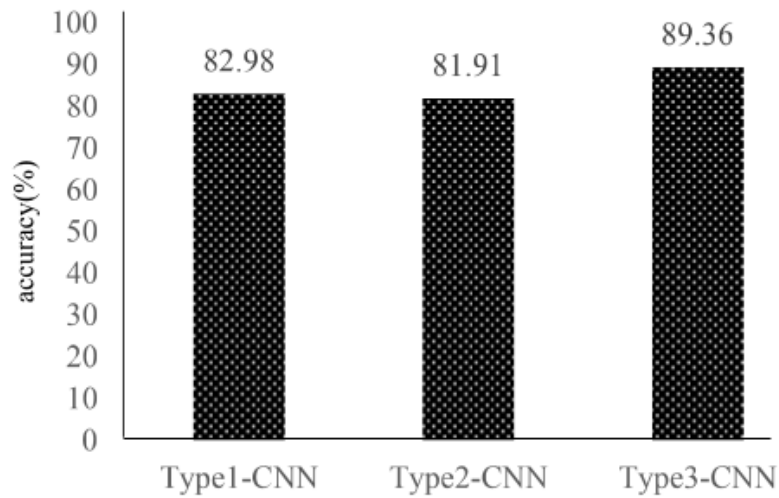


FIGURE 4 – les taux de précision de type1, type2 et type3

TABLE 2 – Architecture

Nome	Donnée	Heure	Taux
Type1	sans pré-traitement	12h	82,98%
Type1	pré-traitement	12h	68,63%
Type2	sans pré-traitement	14h	81,91%
Type2	pré-traitement	14h	74,51%
Type3	sans pré-traitement	16h	89,36%

Les images histopathologiques ont été classe avec succès via les réseau de neurones à convolution. L'architecture type3 a permis d'obtenir des meilleurs résultats, tant dis que l'architecture type1 et 2 avec les images pré-traitée l'algorithme ne donne pas les meilleurs performances.

Dans l'approche[4] Fabio A.sponhol et al ont effectué leurs travail sur les jeux de données de breakhis. L'objectif est de faire l'extraction des caractéristiques avec le Deep Convolutional Activation Feature (DeCAF) en suite utilisé les réseaux de neurones convolutionnel pré-formé disponible gratuitement sur le framework d'apprentissage en profondeur coffe2.

Les chercheurs ont constaté que la plus part des expériences ont été réalisé sur les méthodes artisanales et le CNN. Pour réaliser leur idées ils utilisent deux autres méthodes mais en ce basant sur les réseaux de neurones convolutionnel :

1ère méthode consiste à utilisé le DeCAF en suite utilisé un réseau de neurone convolutionnel pré-formé.

2ème méthode Consiste à réutiliser le fonctionnement de la méthode 1 mais en faisant la combinaison des résultats de classification de sous images grâce au multiples feature vector[4].

D'après les informations la figure 5 on constate que la couche fully connected 8 est moins important par rapport au couches 6 et 7. dans la figure 6 ils ont effectué la combinaison des caractéristiques c'est a dire 6+7+8, 6+7, 6+8, et 7+8. la méthode de combinaison leur a permis d'obtenir les taux de précision supérieur à celui de la méthode 1

	Layer	#p	Magnification factor			
			40×	100×	200×	400×
P	fc8	1	82.0 ± 2.5	82.0 ± 3.6	82.3 ± 2.1	81.3 ± 1.7
		4	82.3 ± 5.5	83.2 ± 6.2	81.6 ± 3.0	79.4 ± 6.3
		16	83.4 ± 6.9	83.8 ± 8.5	85.8 ± 3.5	80.7 ± 9.1
	fc7	1	83.1 ± 2.3	82.6 ± 3.5	82.5 ± 2.3	81.9 ± 2.1
		4	82.7 ± 5.0	83.0 ± 5.9	82.0 ± 2.8	80.4 ± 5.6
		16	83.4 ± 6.7	83.1 ± 8.4	86.0 ± 3.7	81.6 ± 8.6
	fc6	1	82.0 ± 3.3	83.3 ± 4.0	82.4 ± 3.1	81.0 ± 2.5
		4	82.8 ± 5.8	83.9 ± 5.9	81.8 ± 3.8	79.9 ± 6.1
		16	82.5 ± 8.6	83.6 ± 8.5	85.4 ± 5.2	81.1 ± 9.0
I	fc8	1	81.0 ± 1.6	80.9 ± 3.9	81.9 ± 1.1	80.2 ± 1.3
		4	83.7 ± 2.8	84.4 ± 4.3	82.0 ± 1.1	81.0 ± 2.6
		16	83.2 ± 2.4	84.0 ± 4.9	83.4 ± 1.1	80.9 ± 3.7
	fc7	1	82.2 ± 1.4	81.4 ± 3.9	81.9 ± 1.1	80.8 ± 1.5
		4	83.7 ± 2.7	83.7 ± 4.3	82.0 ± 1.1	81.4 ± 2.0
		16	83.1 ± 2.1	83.3 ± 4.6	84.1 ± 1.5	81.6 ± 3.7
	fc6	1	81.1 ± 2.3	82.1 ± 4.0	81.9 ± 1.4	79.8 ± 1.5
		4	84.3 ± 2.9	84.7 ± 4.4	82.2 ± 2.0	81.1 ± 2.2
		16	83.0 ± 2.6	84.6 ± 5.0	84.0 ± 2.8	81.1 ± 3.9

FIGURE 5 – Résultat de la 1ère méthode

	Setup	#p	Magnification factor			
			40×	100×	200×	400×
P	6+7	1	82.6 ± 2.7	83.4 ± 4.4	82.7 ± 2.3	82.1 ± 2.4
		4	83.6 ± 4.9	83.4 ± 6.6	82.1 ± 3.4	80.6 ± 5.7
		16	83.4 ± 7.8	83.6 ± 8.7	85.8 ± 4.2	81.2 ± 9.0
	6+8	1	82.8 ± 2.6	83.5 ± 4.2	83.0 ± 2.1	82.0 ± 2.0
		4	83.4 ± 5.3	83.4 ± 6.6	82.2 ± 3.6	79.9 ± 6.1
		16	83.3 ± 7.4	83.8 ± 8.7	86.0 ± 4.1	80.7 ± 9.2
	7+8	1	82.7 ± 2.3	82.4 ± 3.4	82.8 ± 2.1	82.0 ± 1.5
		4	82.7 ± 5.0	83.3 ± 6.7	81.5 ± 3.2	79.9 ± 6.2
		16	83.3 ± 6.8	83.6 ± 8.6	86.3 ± 3.5	80.8 ± 9.0
I	6+7	1	81.7 ± 1.9	82.4 ± 4.5	82.2 ± 1.0	80.9 ± 1.6
		4	84.6 ± 2.6	84.4 ± 4.7	82.3 ± 1.9	81.5 ± 2.6
		16	83.4 ± 2.3	84.3 ± 5.0	84.1 ± 2.2	81.3 ± 3.9
	6+8	1	81.8 ± 1.8	82.4 ± 4.5	82.6 ± 1.0	80.8 ± 1.3
		4	84.5 ± 3.1	84.8 ± 4.5	82.6 ± 1.7	81.1 ± 2.5
		16	83.2 ± 2.4	84.8 ± 5.1	84.1 ± 2.0	81.0 ± 4.0
	7+8	1	81.8 ± 1.5	81.6 ± 4.0	82.2 ± 1.1	80.9 ± 1.0
		4	83.6 ± 2.8	84.4 ± 4.4	81.7 ± 1.3	81.4 ± 2.4
		16	83.0 ± 2.1	83.9 ± 4.7	84.0 ± 1.3	81.1 ± 3.8

FIGURE 6 – Résultat de la 2ème méthode

4.6 Étude comparative

L'étude comparative consiste à analyser les méthodes appliquées dans les différents articles afin de ressortir les points forts et points faibles ; Dans ce présent document nous allons mener nos études sur l'article 2 et 3.

Méthode	Nombre de couche	Type de données	Type de <u>CNN</u>	Durée	Performance
Supervisé	8	sans	Pre-formé		<u>93% patient</u> <u>90% image</u>
	6	Pre-traité	A partir de zéro	12h	68,63%
		sans		12h	82,08%
	8	Pre-traité	A partir de zéro	14h	74,51%
		sans		14h	81,91%
	10	sans	A partir de zéro	16h	89,36%

FIGURE 7 – Comparaison de méthode

5 Solution proposé

5.1 Contexte général

L'intelligence artificielle est un domaine de l'informatique qui à pour objectif est de doter des systèmes des connaissances similaire à celui des hommes. D'où l'apparition des algorithmes de l'apprentissage automatique permet au système d'acquérir des connaissances à partir des jeux de données de son entraînement. Ces algorithmes présentent des limites pour des tâches de classification de reconnaissance vocale. Pour pousser les limites les chercheurs ont créé un sous domaine de l'apprentissage automatique appelé le deep learning (apprentissage en profond), le deep learning est constitué de plusieurs couches de neurones. La classification d'image consiste à identifier la classe à laquelle l'image appartenant, afin d'identifier la classe d'autre domaine rentre en jeu comme le traitement d'image et la vision par ordinateur. Le thème de notre travail porte sur la classification d'image biologique, vu que ce domaine est très vaste nous avons décidé d'orienter notre réflexion sur la classification des cellules cancéreuses de sein. La raison qui nous a poussé à choisir cette maladie, c'est que le cancer de sein est plus répandu chez les femmes dans le monde en plus il présente un taux de mortalité très élevé par rapport aux autres types de cancer.

5.2 Problématique

L'objectif principal de notre travail est de faire la classification des cellules cancéreuses du sein, elle consiste à déterminer la sortie y à partir de l'entrée x .

5.3 Description de la base de données Breakhis

La base de données Breakhis est composée de 7909 images microscopiques de sein recueillies auprès de 82 patients en utilisant différents facteurs d'agrandissement (40x, 100x, 200x et 400x). Au total, il y a 2480 échantillons de bénignes et 5429 échantillons de malignes ; C'est deux types de tumeurs peuvent être regroupés en fonction de l'aspect des cellules. Dans notre jeu de données nous constatons quatre types histologiques distincts de tumeur bénigne et aussi de maligne à savoir :

Bénigne

- Adenosis (B_A), Fibroadenoma (B_F), Phyllodes tumor (B_PT), Tubular adenoma (B_TA) ;

Maligne

- Ductal carcinoma (M_DC), Lobular carcinoma (M_LC), Mucinous carcinoma (M_MC), Papillary carcinoma (M_PC).

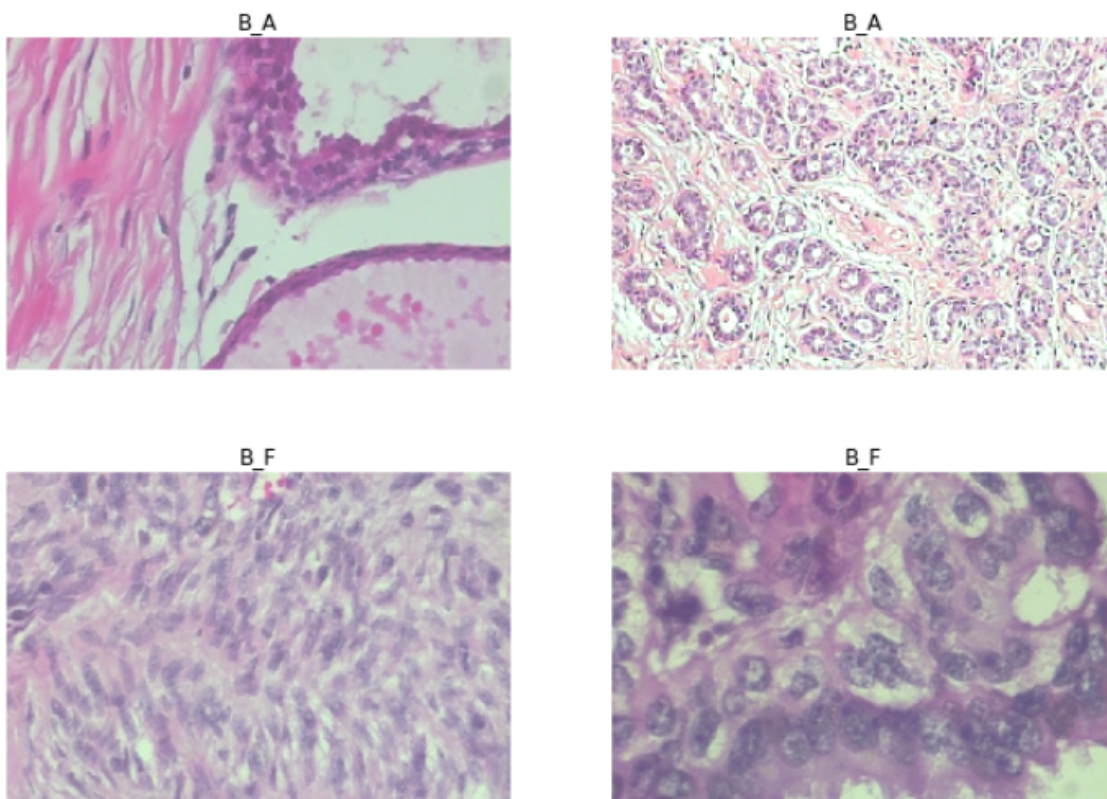


FIGURE 8 – *Tumeur Bénigne*

5.4 Solution proposée

La solution proposée est une étape important pour la suite de travail. Car la méthode choisie sera implémenté au niveau de la partie pratique.

D'après les études effectués au niveau de l'état de l'art nous avons constaté que la méthode utilisé dans[3] est le mieux adopter à notre problème

5.5 Contribution

Après avoir crée un réseau de neurone convolutionnel à partir de zéro, nous envisageons comme contribution une stratégie basé sur l'apprentissage par transfert

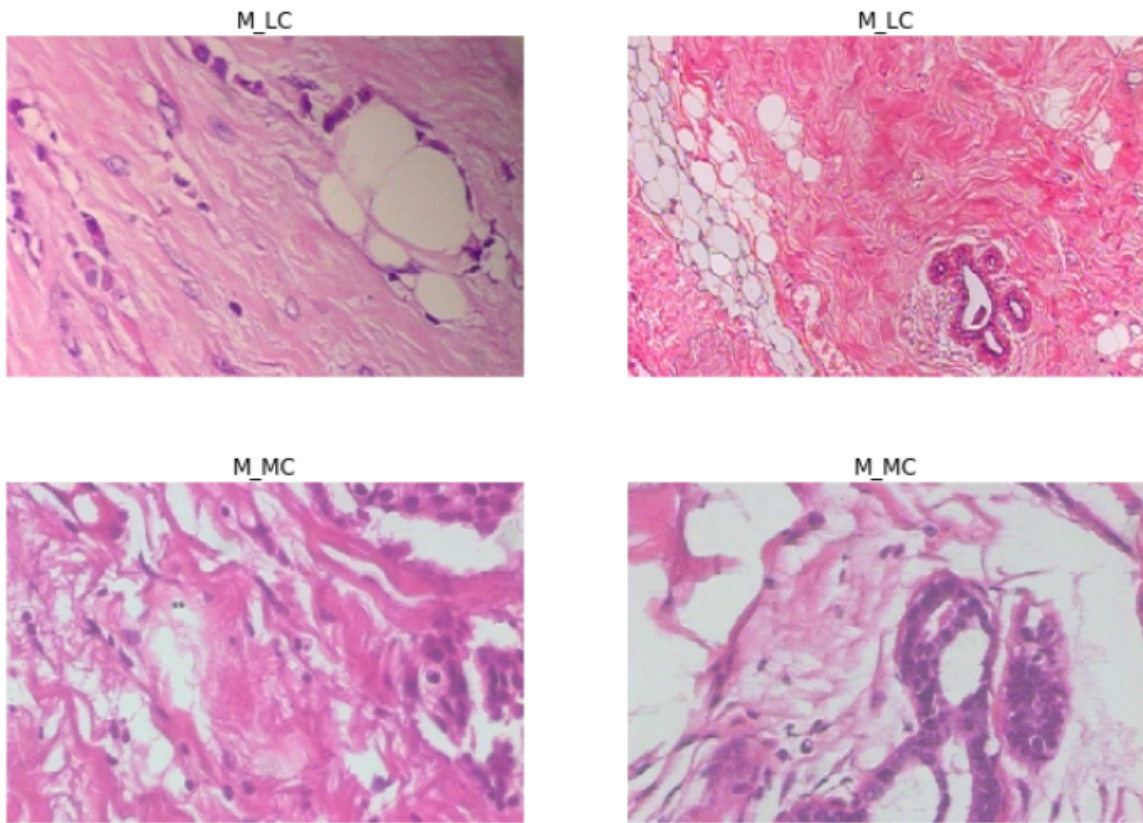


FIGURE 9 – *Tumeur Maligne*

5.6 Présentation des outils

5.6.1 Le software

Il existe plusieurs frameworks open sources pour créer les réseaux profonds, Mais la majorité supporte le langage python. Voici la liste de quelques-uns d'entre eux :

5.6.2 Theano

Créé par Frédéric Bastien et l'équipe de recherche derrière le laboratoire de l'Université de Montréal.

Avantages :

- Python
- Performant si utilisé correctement

Inconvénient :

- Les grands modèles peuvent nécessiter un long temps de compilation
- API bas niveau

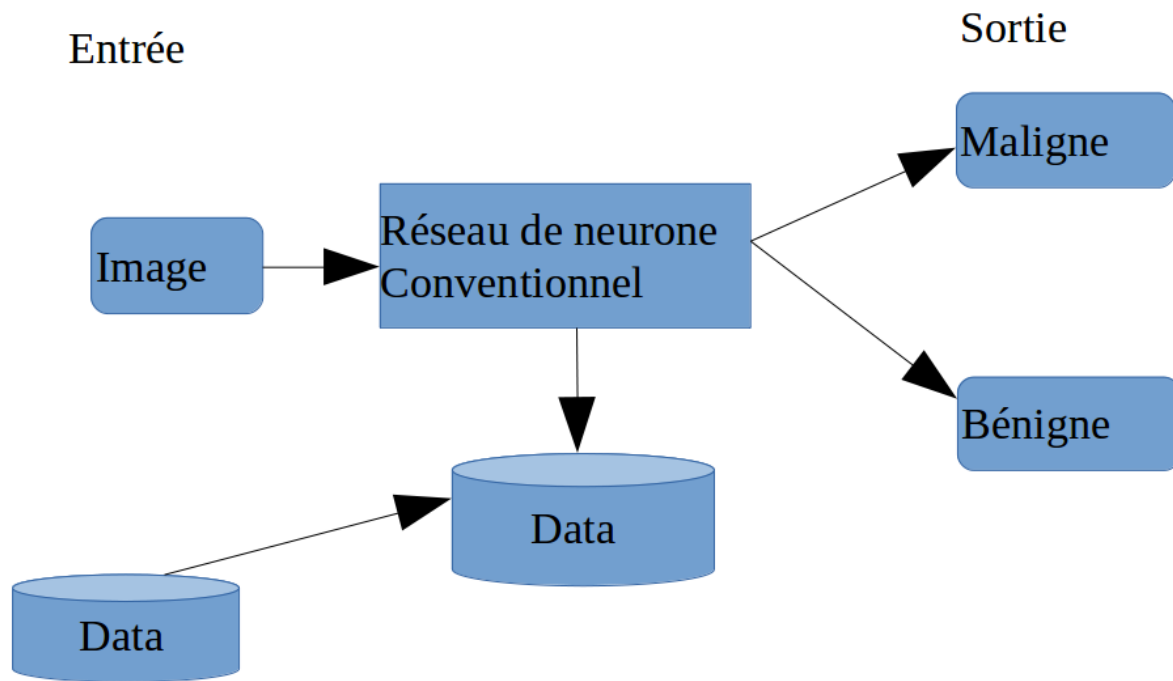


FIGURE 10 – Architecture du Prototype à développer

5.6.3 Keros

Le framework le plus haut niveau, le plus convivial de la liste. Il permet aux utilisateurs de choisir si les modèles qu'ils construisent sont exécutés sur Theano ou TensorFlow. Il est écrit et entretenu par Francis Chollet, un autre membre de l'équipe Google Brain.

Avantages :

- Python
- Le backend par excellence pour Theano ou TensorFlow
- Interface haut niveau, intuitive

Inconvénient :

- Moins flexible que les autres API

5.6.4 Autres frameworks

- TensorFlow : par Google
- CNTK : par Microsoft
- Neon : par Nervana Systems. Il a récemment été classé comme le framework le plus rapide dans plusieurs catégories
- Deeplearning4j : Il supporte le langage java
- Caffe : par Berkeley Vision and Learning Center

5.7 Outils à utiliser

5.7.1 Hardware

L'apprentissage en profondeur est un domaine très exigeant au niveau des ressources, Car il effectue des milliers voir plus lors de la formation ou au niveau de la détermination des nouvelles caractéristiques. Pour cela nous avons décidé d'effectuer les expérimentations sur un ordinateur hp EliteBook acceptable avec les caractéristiques suivantes :

CPU	Intel Core i5-33(2,40GHZ)
RAM	8GB

5.7.2 Software

TensorFlow est un système d'apprentissage automatique, Il se présente comme une bibliothèque dédiée au calcul numérique . le but était d'optimiser les calculs numériques complexes, mais aujourd'hui il est très connu pour résoudre des problèmes de Deep learning.

Avantages :

- Python
- Multi-plateforme(Linux,MAC OS,Android,IOS)
- APIs en python,C++,java
- supporté par Google
- Une très grande communauté
- Le support les calculs sur CPU,GPU et

Inconvénient :

- Plus lent que les autres framework dans de nombreux benchmarks, bien que Tensorflow se rattrape.

6 Planification des tâches

Travaux	Durée estimée
Installation d'outils	1 semaine
Mise en place du prototype	6 semaine
Apprentissage,Test, Amélioration	2 semaine
Test de validation du prototype	2 semaine

TABLE 3 – Planification tâche

7 Expérimentation

7.1 Nation de base

Avant même la présentation de l'architecture du model, Il est très judicieux de commence par expliquer les paramètres de la couche convolutive enfin de déterminer les dimensions spatiales de sortie de chaque couche.

La formule de sortie de chaque couche convolutive est :

$$\text{Output} = \frac{N-F}{S} + 1 \text{ avec}$$

N : Taille des entrées

F : Taille du filtre

S : le pas

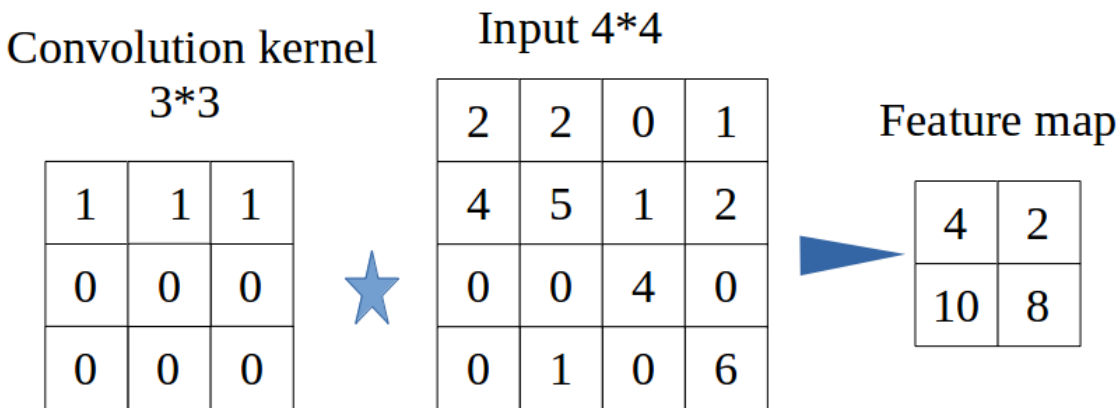


FIGURE 11 – zero padding

D'après la figure, l'image d'entrée est de taille 4x4, dimension du filtre 3x3, supposons que le pas $S=1$. A la sortie nous aurons une image de taille : $\text{Output} = \frac{4-3}{1} + 1 = 2$

Dans le même cas de figure lorsque nous avons changé la taille du pas, on a constaté que la dimension de l'image est continue ce qui n'est pas possible.

Prenons l'exemple ou le pas $S = 3$, La sortie $\text{output} = \frac{4-3}{3} + 1 = 0.66$

Nous obtenons à la sortie une image de taille 0.66×0.66 , ce qui n'est pas possible.

Pour résoudre ça, en pratique on applique la méthode Non-Zero Padding. Le Non-Zero Padding permet d'ajouter des zéros au bordures de l'image afin d'avoir la même taille d'image à la sortie de la couche convolutive.

Sa formule est déterminé comme suit : $P = \frac{F-1}{2}$

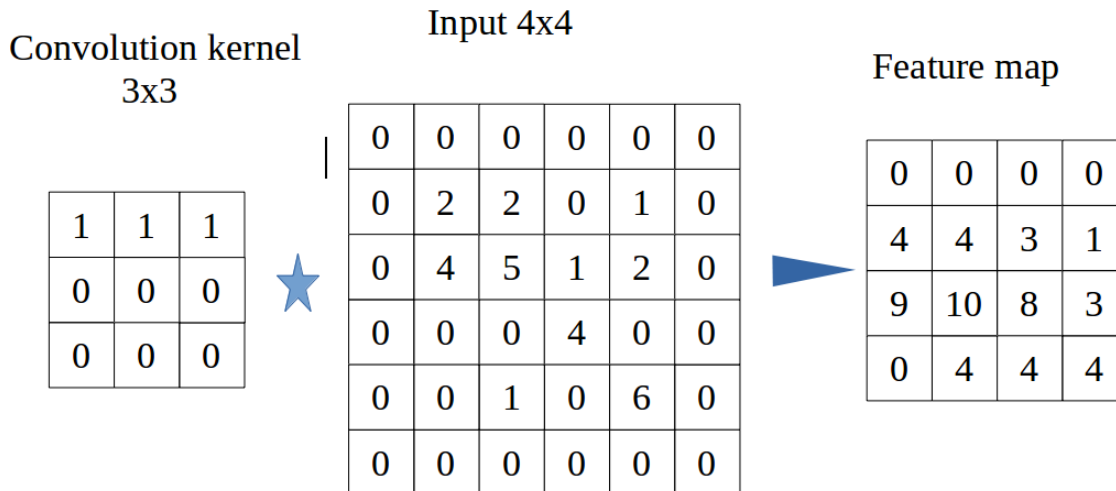


FIGURE 12 – Non zero padding

D'après la formule de pas, $P = \frac{3-1}{2} = 1$. Cela veut dire que nous allons ajouter des pixels d'intensité nulle surtout le long de la bordure de l'image sur un seul niveau. La nouvelle dimension de l'image sera de 6x6.

A partir de maintenant nous pouvons alors déterminer la taille de sortie de l'image ainsi que le nombre de paramètre par couche convolutive.

Soit a la entrée une image de dimension 100x100x3(3 pour le canal RGB). On décide d'appliquer une couche convolutive de dimension $F = 5$ et un nombre de filtres $k = 16$ et un stride $S = 1$. La première étape est d'appliquer la méthode non zero padding :

$$P = \frac{5-1}{2} = 2$$

La nouvelle taille de l'image est de $100 + 2 \times 2 = 104$. Avec la formule $\frac{N-F}{S} + 1$, Nous pouvons calculer la taille de l'image a la sortie :

$$\text{Output} = \frac{104-5}{1} + 1 = 100$$

Vus que nous avons 16 filtres dans la couche convolutive alors nous obtiendrons 100x100x16.

Pour calculer le nombre de paramètre de la couche : $5 \times 5 \times 3 + 1 = 76$ (5×5 taille du filtre, 3 pour le nombre de canal de l'entrée, 1 pour le biais), puis que la couche possède 16 filtres alors le nombre de paramètres de la couche est : $76 \times 16 = 1216$.

8 Architecture

Après l'implémentation du modèle, la première des choses c'est l'apprentissage du réseau de neurones convolutif. Le réseau possède 4.796.776 paramètres entraînaables et il consiste en 6 couches convolutives, une couche de Dropout est placée après chaque couche convolutive, une couche de Maxpooling est placée après chaque série de 2 couches convolutives et enfin un MLP à 3 couches



FIGURE 13 – zero padding

9 Initialisation des paramètres

L'application des bons poids est recommandé lorsque l'on souhaite que le réseau fonctionne correctement. Il est important de s'assurer que les poids soient dans une intervalle raisonnable avant d'entraîner le réseau. Pour cela, nous utiliserons **Glorot initialization**.

9.1 Glorot initialization

fixer les valeurs du poids du réseau avant de lancer l'apprentissage paraître être un processus aléatoire. nous n'avons donc pas la sûreté d'avoir attribuer les bons poids qui fonctionneraient dans ce cas particulier. La meilleur façon d'assigner les poids est d'appliquer la distribution gaussienne. Évidemment, cette distribution aurait une moyenne nulle et une variance finie. Considérons un neurone linéaire :

$$y = w_1x_1 + w_2x_2 + \dots + w_nx_n + b$$

Avec x l'entrée constituée de n composantes. Avec chaque couche qui passe, nous voulons que la variance reste la même. Cela nous aide à empêcher le signal d'exploser à une valeur élevée ou à disparaître à zéro. En d'autres termes, nous devons initialiser les poids de telle sorte que la variance reste la même pour x et y . Ce processus d'initialisation est connu sous le nom de Glorot initialization.

10 Régularisation

le grand problème du machine learning (ML) est de savoir créer un algorithme qui répond bien sur l'ensemble d'apprentissage et aussi sur l'ensemble de test. De nombreuses stratégies utilisées dans le Deep Learning sont explicitement conçues pour réduire l'erreur de test, éventuellement au détriment d'une augmentation de l'erreur d'apprentissage. Ces stratégies sont connues sous le nom de régularisation. Nous allons nous intéresser à une technique particulièrement puissante : **le Dropout**.

10.1 Dropout

Le Dropout est une technique où des neurones sélectionnés au hasard sont ignorés (temporairement) pendant l'apprentissage. Cela signifie que leur contribution à l'activation des neurones qui leur succède est temporairement supprimé lors de la phase de propagation et toutes les mises à jour de poids ne sont pas appliquées au neurone lors de la phase de retro propagation.

Lorsque des neurones sont supprimés au hasard du réseau pendant l'apprentissage, les autres neurones devront intervenir et gérer la représentation requise pour faire des prédictions pour les neurones manquants.

Lors de la phase d'apprentissage, pour chaque itération, un neurone est gardé avec une probabilité p , sinon il est supprimé.

Lors de la phase de test, tous les neurones sont gardés, nous voulons donc que les sorties des

neurones au moment du test soient identiques à leurs sorties au moment de l'apprentissage. Par exemple, dans le cas où $p = 0.5$, les neurones doivent réduire de moitié leurs sorties au moment du test pour avoir la même sortie que pendant l'apprentissage.

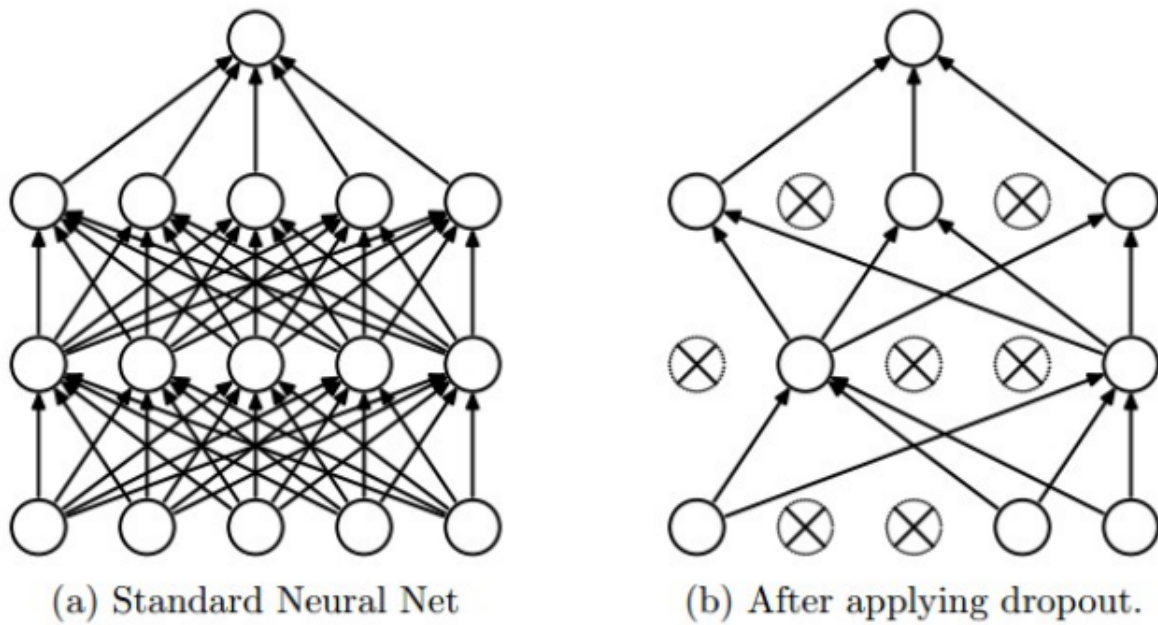


FIGURE 14 – Dropout

11 Résultat de l'apprentissage

Après l'assignation du poids, l'étape suivante est l'apprentissage. Le but est d'apprendre le réseau à reconnaître les différences qui existent entre nos deux types d'image **Maligne et Bénigne**.

Vu que la base de données est composée des images de taille (40,100,200,400), Nous avons réalisé l'apprentissage sur chaque des tailles afin de voir celui qui offre le meilleur taux d'apprentissage.

le tableau ci-dessous récapitule les résultats obtenus au cours de l'apprentissage

	40x40	100x100	200x200	400x400
Résultat article	88%	88.8%	88.7%	86.7%
Résultat cas pratique	42%	38%	41%	42%

FIGURE 15 – *Taux apprentissage*

D'après les résultats obtenus ci-dessus, nous voyons clairement que la performance du réseau mise en place est en dessous des résultats obtenus dans [4]. Pour pouvoir améliorer la performance du réseau nous allons appliquer d'autres architectures en changeant les paramètres

12 Conclusion et perspectives

L'apprentissage profond représente l'une des techniques les plus importantes. Du fait que les autres techniques ont montré leurs limites dans certains domaines comme la traduction de langue, la classification d'objets.

Dans ce travail, nous avons passé la majeure partie de notre temps à lire et à analyser des articles et les publications afin de mieux comprendre les modèles utilisés jusqu'à présent. Cela nous a permis de concevoir notre propre modèle de classification d'image.

Comme perspectives nous pouvons citer :

- Test sur de nouvelles bases
- Segmentation et augmentation d'image.

Références

- [1] reseaux de neurone artificiels. <https://hal-amu.archives-ouvertes.fr/hal-01338010/document>. [Online ; accessed 21-Mars-2018].
 - [2] Carlos Affonso, André Luis Debiaso Rossi, Fábio Henrique Antunes Vieira, André Carlos Ponce de Leon Ferreira, et al. Deep learning for biological image classification. *Expert Systems with Applications*, 85 :114–122, 2017.
 - [3] Toto Haryanto, Ito Wasito, and Heru Suhartanto. Convolutional neural network (cnn) for gland images classification. In *Information & Communication Technology and System (ICTS), 2017 11th International Conference on*, pages 55–60. IEEE, 2017.
 - [4] Fabio A Spanhol, Luiz S Oliveira, Paulo R Cavalin, Caroline Petitjean, and Laurent Heutte. Deep features for breast cancer histopathological image classification. In *Systems, Man, and Cybernetics (SMC), 2017 IEEE International Conference on*, pages 1868–1873. IEEE, 2017.
- <https://fr.mathworks.com/discovery/deep-learning.html>. "[Online ; accessed 21-Mars-2018]"
- https://fr.wikipedia.org/wiki/Apprentissage_profond "[Online ; accessed 19-Mars-2018]"
- http://www.lemonde.fr/pixels/article/2015/07/24/comment-le-deep-learning-revolutionne-l-intelligence-artificielle_469592_4408996.html "[Online ; accessed 19-Mars-2018]"
- https://fr.wikipedia.org/wiki/Apprentissage_profond "[Online ; accessed 10-Mars-2018]"
- <http://neuralnetworksanddeeplearning.com/chap6.html> "[Online ; accessed 16-Mars-2018]"