Mémoire :
Reconnaissance d'émotions par l'expression faciale

Majeure : Intelligence Artificielle

Equipe :

- M.Lucas Barrot
- M.Simon Garras

Référent :

- M. Larbi Boubchir

Avril - Septembre 2020

Table des matières

1. Introduction

1.1. Objectif et équipe

L'objectif principal de ce projet sera de mettre en œuvre une application d'intelligence artificielle qui permettra, à partir de la photo du visage d'une personne, de reconnaître de manière automatique son état émotionnel.
Notre groupe est composé de Lucas Barrot et Simon Garras. Nous sommes tous deux étudiants en 4ème année d'école d'ingénieur, à l'ESME Sudria, dans la majeure Intelligence Artificielle.
Ce projet se place donc dans le contexte de la majeure *intelligence artificielle* dans les branches machine learning et traitement d'image, dans le but d'approfondir nos compétences dans ces deux domaines.

1.2. Informations transmises par le visage

Le visage est porteur de plusieurs informations importantes dont les deux principales sont l'identité et les expressions faciales. Nous ne nous intéresserons ici qu'au expressions faciales car c'est le sujet de notre projet.
Les expressions faciales sont un des moyens les plus utilisés pour transmettre les états émotionnels qui jouent un rôle fondamental dans les interactions entre les personnes.
Les expressions sont déterminées par l'activation des muscles faciaux (comme illustré avec la figure 1.2.1). De plus, certains processus émotionnels peuvent faire changer localement la couleur de la peau, en la colorant localement par un afflux sanguin plus important qu'à l'accoutumée.

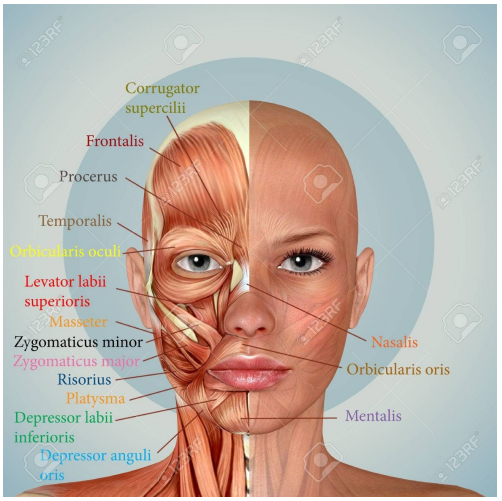


Figure 1.2.1 : Muscles faciaux

C'est à ces expressions, leur nature et les informations qu'elles transmettent que nous allons nous intéresser dans ce projet.

1.3. Problématique

La problématique de ce projet est le développement d'un outil de reconnaissance d'émotions faciales grâce à un algorithme d'intelligence artificielle. Il faut donc choisir un certain nombre d'émotions communes à tout humain et les donner à un algorithme afin qu'il puisse les reconnaître avec le maximum de précision possible.

1.4. Domaines d'applications

Apprendre à reconnaître l'état émotionnel d'une personne peut rendre possible la simulation de ses états sur une machine et améliorer l'interaction entre l'homme et la machine.

On peut également imaginer d'autres domaines d'application comme dans la sécurité ou encore le commerce. En effet, reconnaître un visage et son expression peut également amener à interpréter ses actions présentes (et potentiellement futures) à des fins de prévention (risque terroriste, fraude, fatigue ou manque d'attention en voiture), à des fins commerciales (Avoir directement le retour du client sur un produit en analysant son expression faciale) ou simplement publicitaires (On peut imaginer une publicité adaptée à l'émotion perçue d'un potentiel client).

1.5. Plan du mémoire

Nous allons détailler dans le prochain chapitre dans un premier temps comment fonctionnent les émotions et comment on peut les reconnaître et les interpréter, puis faire un état de l'art des méthodes de reconnaissance émotionnelle déjà existants en s'intéressant plus particulièrement aux méthodes utilisant le deep learning.

2. Reconnaissance des émotions

Avant de pouvoir parler de la reconnaissance des émotions, il faut revenir à ce qu'est une émotion. Nous allons ensuite parler des grandes théories servant à catégoriser les émotions

2.1. L'émotion

L'émotion a été défini sous quatre grands aspects : physiologique, subjectif, cognitif et expressif.

L'aspect physiologique des émotions se manifeste par les changements corporels accompagnant le changement de l'état subjectif comme le changement de la température corporelle, de la fréquence cardiaque et respiratoire. La détection de ces aspects nécessite des mesures avec des capteurs liés aux sujets, ce qui s'avère être intrusif.

L'aspect subjectif des émotions qui est lié au ressenti par les individus et l'aspect cognitif qui est lié à la compréhension de l'individu sur une scène ou un événement.

L'aspect expressif qui est constitué des expressions faciales, des expressions corporelles et des intonations vocales.

Notre projet s'inscrit dans le cadre d'une reconnaissance des émotions à travers leurs aspects mesurables à savoir ici non pas l'aspect physiologique (donc sans l'utilisation d'outils intrusifs) ou l'aspect subjectif ou cognitif (qui relèvent plus de la psychologie), mais bien de l'aspect expressif

Il faut à présent dresser une liste des émotions "affichable" par un visage et (plus important) les catégoriser.

2.2. Séquencement des émotions

De nombreuses études se sont portées sur les émotions et leurs séquencement.

On peut noter la théorie de l'universalité des émotions qui suppose que l'émotion est universelle, dotée de fonctions adaptatives et de fonctions communicatives. Elle a été exposé par Darwin dans son livre "L'expression des émotions chez l'homme et les animaux".

Selon sa théorie, les émotions sont imprimées dans le système nerveux humain, les rendant universelles. Il classe les changements des expressions faciales en sept groupes pour lesquels les déformations du visage y sont décrites de façon détaillée.

Cette théorie a engendré de nombreuses études qui ont tenté de la prouver en travaillant sur un nombre restreint d'émotions renommées émotions de base (émotions discrètes, émotions primaires ou émotions fondamentales).
D'après Ekman, les émotions de base ont des caractéristiques uniques et leurs réactions sont préprogrammées. De plus, elles sont souvent représentées comme des émotions innées et indépendantes des cultures. Ekman définit sept autres critères pour caractériser les émotions fondamentales [3], tels que la présence de la même émotion chez un autre primate ou la présence de caractéristiques physiologiques qui la distinguent des autres émotions

La théorie de l'universalité a facilité la représentation des émotions, notamment la représentation discrète (catégorielle).

La représentation discrète catégorise certaines émotions dans des groupes prédéfinis ayant des caractéristiques distinctes les unes des autres. Ce sont les émotions de bases qui symbolisent ces catégories.

Les chercheurs qui adoptent le concept des émotions de base présentent un ensemble restreint d'émotions, mais leurs recherches divergent lorsqu'il s'agit de les identifier. Les travaux ci-dessous présentent les émotions de base selon chaque auteur :

- Ekman : colère, peur, tristesse, joie, dégoût, surprise.
- Tomkins : colère, dégoût, joie, peur, surprise, mépris, honte, intérêt, anxiété.
- Izard : colère, surprise, dégoût, joie, peur, tristesse, mépris, intérêt, culpabilité, honte.

Un autre courant de pensées représente les émotions de manière continue (dimensionnelle) basé sur deux ou plusieurs dimensions.

Contrairement à la représentation catégorielle, les émotions sont définies en se basant seulement sur les dimensions. Russell définit les états affectifs sur un modèle bidimensionnel, représenté par un cercle basé sur un axe de plaisir (plaisir/peine) et un axe quantifiant la force du ressenti (illustré sur la Figure 2.2.1).

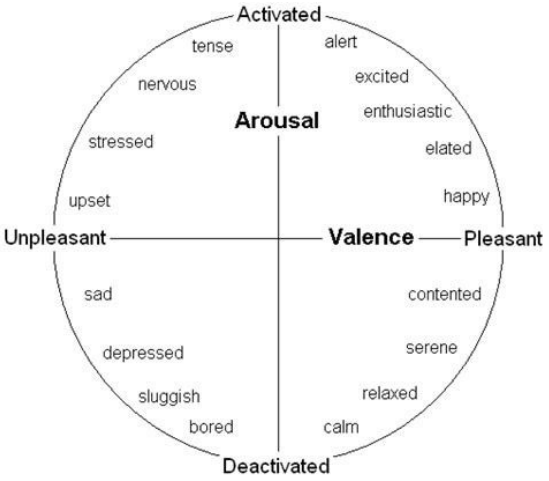


Figure 2.2.1 : Modèle circulaire de Russell

Toutes les émotions peuvent être définies sur ce modèle. Si on prend l'exemple de l'enthousiasme, il s'agit d'un état associant un taux de plaisir élevé à un taux d'activation élevé. D'après Russell, c'est un état qui se trouve à un angle de 45° en considérant que le plaisir est l'angle 0° et l'état d'éveil est l'angle 90°.
D'autre part, Cowie et al proposent un outil pour l'annotation des émotions sur un modèle bidimensionnel très proche de celui de Russell, qui se base sur l'activation et l'évaluation (positif-négatif).

Le modèle de Plutchik une autre manière de représenter les émotions. Huit émotions de base (colère, dégoût, peur, joie, tristesse, surprise, acceptation et anticipation) sont représentées sur un cercle (sans pour autant les définir par des dimensions). Des émotions secondaire sont définies par des combinaisons des émotions de base adjacentes. Une dimension d'intensité détermine également l'intensité des émotions de base (Figure 2.2.2).

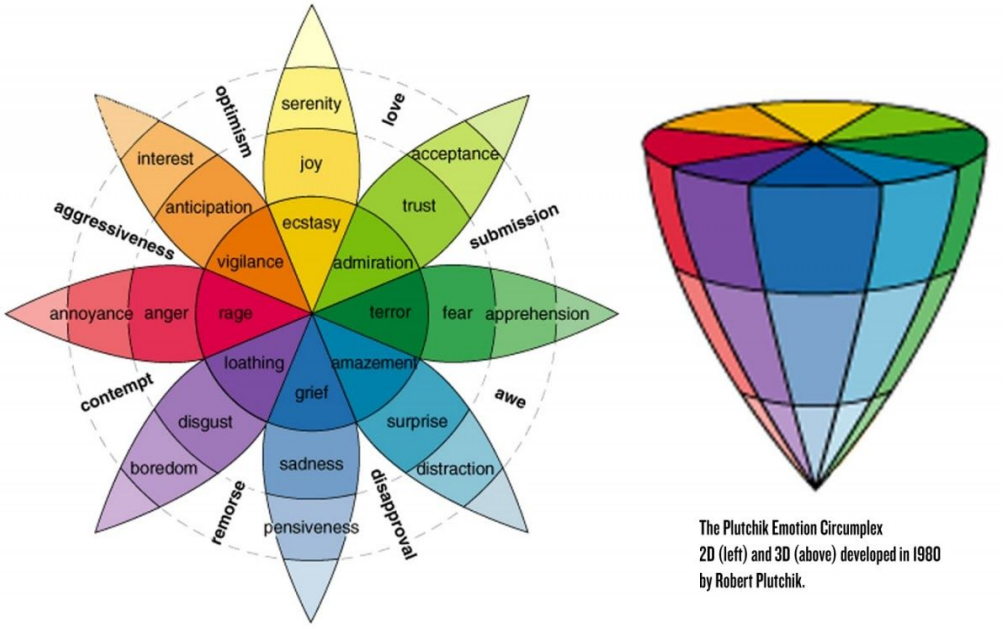


Figure 2.2.2 : Modèle de Plutchik

2.3. Architecture typique d'une application d'un système de reconnaissance faciale d'émotions :

Si on prend l'exemple d'une application mobile :

Une architecture typique commence par un programme de réception (envoie de l'image ou réception après prise de photo par le téléphone). Puis, l'image est envoyé à une API qui la traite de manière à ce qu'elle respecte des propriétés particulières pour qu'elle puisse être insérée dans l'algorithme final (format, couleur ou non, focus sur le visage, etc…).

L'image mise en forme est donnée en entrée à l'algorithme de reconnaissance d'émotion et renvoie l'émotion de sortie du programme.

3. État de l'art: apprentissage profond

Nous avons choisi de diviser ce chapitre en deux, car nous nous sommes rendu compte que les méthodes les plus efficaces entraient dans le cadre du deep learning.
Cependant, il est intéressant de parler de méthodes (d'une en particulier) qui n'entrent pas dans ce cadre-là et qui ont déjà été utilisées pour faire de la reconnaissance d'émotion par l'expression faciale.

3.1. Sans deep learning : La méthode SVM (séparateurs à vaste marge)

Les SVM ont été développés dans les années 1990 à partir des considérations théoriques de Vladimir Vapnik sur le développement d'une théorie statistique de l'apprentissage : la théorie de Vapnik-Chervonenkis. Ils ont rapidement été adoptés pour leur capacité à travailler avec des données de grandes dimensions et le faible nombre d'hyper-paramètres.

Les SVM sont utilisés pour résoudre des problèmes de discrimination, c'est-à-dire décider à quelle classe appartient un échantillon, ou de régression, c'est-à-dire prédire la valeur numérique d'une variable.

La justification intuitive de cette méthode d'apprentissage est la suivante : si l'échantillon d'apprentissage est linéairement séparable, il semble naturel de séparer parfaitement les éléments des deux classes de telle sorte qu'ils soient le plus loin possible de la frontière choisie.
Pour résoudre un problème de classification à deux classes, il s'agit de trouver l'hyperplan le plus éloigné des points les plus proches de chaque classe. Ce qu'on appelle « l'hyperplan séparateur optimal »

Un réseau SVM ne résout qu'un problème de classification binaire. Lorsqu'on traite un problème multi-classes, telle que la reconnaissance des visages, une combinaison des SVM est accommodée.

3.2. Avec Deep learning

3.2.1. L'ajout du deep learning par rapport au supervisé

La méthode support vector machine (SVM) est utilisée principalement pour des problèmes de classification et de régression. La plupart du temps, la SVM sera utilisée sur des jeux de données réduits et des problèmes linéaires la où les méthodes deep learning et plus particulièrement les réseaux de neurones convolutionnels (CNN) vont prouver leur efficacité en utilisant des jeux de données très grands et sur des problèmes non-linéaires.

Le plus gros avantage du deep learning est donc sa capacité d'adaptation à des problèmes plus complexes et moins linéaires, en effet, on peut augmenter la complexité d'un réseau de neurones en ajoutant des couches à l'intérieur du réseau, alors qu'en SVM on ne peut pas du tout augmenter la complexité du modèle, cela se traduit néanmoins par un temps d'entraînement plus long et plus coûteux en ressource.

Cependant dans notre contexte une méthode SVM sera donc insuffisante car notre problème est très complexe et a beaucoup de paramètres à prendre en compte, la taille et la forme du visage, les micro-expressions varient en fonction des personnes et de plus on cherche à reconnaître une émotion parmi 6. La tâche est donc très complexe et non-linéaire, c'est pourquoi nous avons choisi d'utiliser une méthode deep learning.
Le prochain chapitre sera donc sur le deep learning et ses différentes méthodes.

3.2.2. Les différents réseaux neuronaux :

Le deep learning ou l'apprentissage profond utilise les réseaux de neurones artificiels pour résoudre des problèmes généralement complexes et non-linéaires

RNN

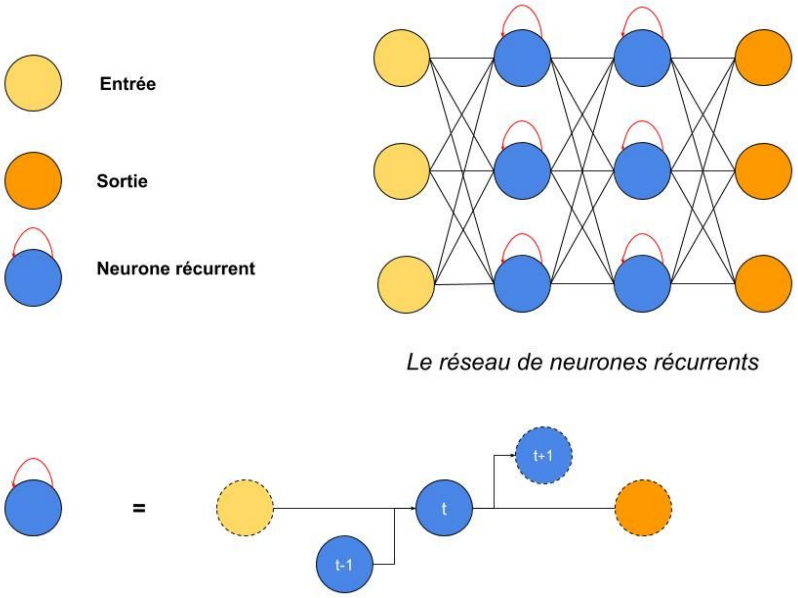
Les réseaux de neurones récurrents (RNN) sont largement utilisés en intelligence artificielle dès lors qu'une notion temporelle intervient dans les données.
Ils sont largement utilisés dans l'analyse de texte (prédiction, traduction, reconnaissance automatique), dans l'analyse vocale ou encore dans la reconnaissance de formes. ils sont très similaires aux réseaux de neurones artificiels (ANN) :

Un ANN fonctionne comme tel (voir Figure 3.2.2.1) :

Des données d'entrées (input) arrivent dans la couche d'entrée du réseau. Les données sont sous la forme d'un vecteur, par exemple (0.7, 0.4, 0.9) et chaque coordonnée du vecteur est envoyée aux neurones d'entrée (jaunes).
Ensuite, les 3 valeurs vont avancer dans le réseau couche par couche (1ère bleue puis 2nde bleue puis orange qui est la sortie). Les couches bleues sont dites cachées et la orange dite couche de sortie.
Pour avancer, entre chaque neurone il y a un trait qui les relie : ce trait est associé à une valeur dite le poids (par exemple 0.3 entre le 1er neurone jaune et le 1er bleu) qui va

pondérer la valeur entrante (i.e. on aura 0.7*0.3 = nouvelle valeur qui arrive dans le 1er neurone bleu).
Toutes les valeurs entrantes (et pondérées) sont additionnées à l'entrée d'un neurone puis on applique une certaine fonction au résultat, ce qui donne une valeur de sortie pour chaque neurone.
Ces valeurs sont ensuite propagées à la couche suivante et ainsi de suite.

Contrairement à un ANN (et c'est là la seule différence), sur chaque neurone bleu, on a une boucle (développée sous le schéma du réseau) :
On a en entrée un vecteur composé de 3 valeurs, une pour chaque neurone jaune, elles se propagent ensuite dans le réseau à la manière d'un ANN, sauf que chaque neurone bleu reçoit en plus des sorties des neurones de la couche précédente, sa propre sortie si il avait en entrée la valeur que le neurone précédent a eu en entrée. de plus sa valeur de sortie est conservée et servira pour le prochain neurone.
On a donc une mémoire de 1 itération pour les neurones bleus.



Le réseau de neurones récurrents

Figure 3.2.2.1 : Schéma représentatif d'un RNN

Cette méthode souffre néanmoins d'un problème majeur qui est son temps d'apprentissage, en effet, comme beaucoup d'autres méthodes, le calcul et la mise à jour des poids se fait grâce à la méthode de rétropropagation du gradient, or en RNN la fonction la plus utilisée pour ce calcul est la fonction tanh qui donne en sortie une valeur entre 0 et 1. On obtient donc très vite plus on recule dans le réseau des valeurs proches de 0, ce qui signifie que les neurones des couches précédentes ont très vite un impact très faible sur la modification des poids, le réseau peut donc facilement "oublier" des données prises en compte plus tôt, on dit qu'il a la mémoire courte.

CNN

Les réseaux de neurones convolutifs ou réseau de neurones à convolution (en anglais CNN ou ConvNet pour Convolutional Neural Networks) sont des réseaux de neurones artificiels acycliques (feedforward), dans lequel le motif de connexion entre les neurones est inspiré par le cortex visuel des animaux. Les neurones de cette région du cerveau sont arrangés de sorte qu'ils correspondent à des régions qui se chevauchent lors du pavage du champ visuel. Leur fonctionnement est inspiré par les processus biologiques, ils consistent en un empilage multicouche de perceptrons, dont le but est de prétraiter de petites quantités d'informations.

Un réseau de neurones CNN est composé principalement de deux types de neurones organisés en couches qui traitent successivement l'information :

- Les neurones de convolution :
  Ces neurones ont pour travail d'effectuer un traitement convolutif sur une portion de l'image limitée au moyen d'une fonction de convolution.
- Les neurones de mise en commun :
  Ces neurones, dits de "pooling" ont pour travail de regrouper les différentes portions de l'image après qu'elles aient été traitées par le noyau de convolution afin qu'elles soient transmises à la couche suivante.

Une architecture CNN est composée d'un empilement de couche de traitements telles que :

- la couche de convolution (CONV) qui traite les données d'un champ récepteur ;
- la couche de pooling (POOL), qui permet de compresser l'information en réduisant la taille de l'image intermédiaire (souvent par sous-échantillonnage) ;
- la couche de correction (ReLU), souvent appelée par abus « ReLU » en référence à la fonction d'activation (Unité de rectification linéaire) ;
- la couche « entièrement connectée » (FC), qui est une couche de type perceptron ;
- la couche de perte (LOSS).

Ci-dessous un schéma explicatif d'une architecture CNN très commune.

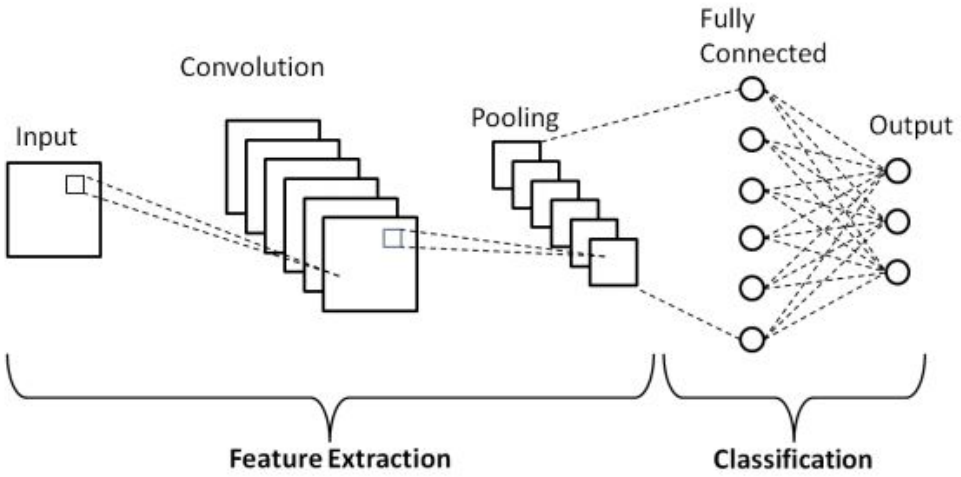


Figure 3.2.2.2 : Schéma représentatif d'un CNN

Le principal avantage des CNN pour la résolution de notre problème est que ceux-ci sont très adaptés aux problèmes non-linéaires tels que la reconnaissance d'émotions et peuvent augmenter en complexité "simplement" en rajoutant des couches intermédiaires afin d'avoir un modèle plus profond et plus performant.
Il existe des centaines d'architectures différentes de CNN, nous allons donc en choisir quelques unes et les entraîner sur notre jeu de données afin de voir laquelle est la plus adaptée et nous renvoie les meilleurs résultats.

Auto encoding

De manière général : le principe d'un auto encodeur est d'apprendre à compresser et encoder les données puis apprendre à les reformer le plus fidèlement possible à partir de la forme compressée (comme illustré dans la **figure 3.2.2.2)** .
De cet manière, l'auto encodeur apprend à ignorer le bruit.

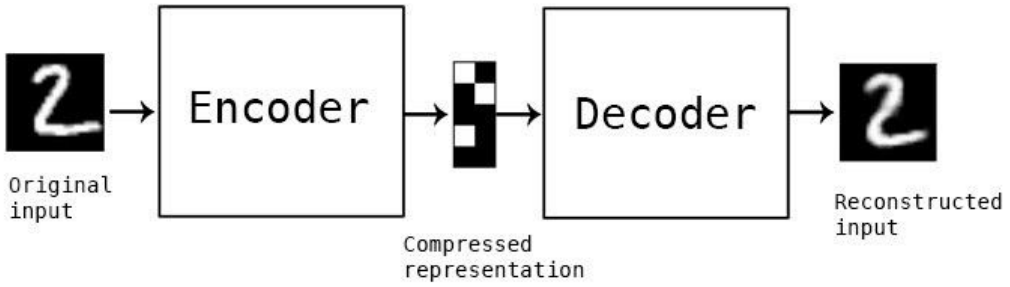


Figure 3.2.2.2 : Schéma illustratif du fonctionnement d'un auto-encodeur

Ce réseau de neurones est très utilisé pour la détection et la réduction de bruit **(figure 3.2.2.3)** ou la détection d'une anomalie.

Le réseau de neurones d'un auto encodeur est constitué de 4 grandes parties

1. L'encodeur : Dans cette partie, le model apprend à réduire les dimensions de la donnée d'entrée et la compresse dans une forme dite "encodée".
2. Le Goulot : qui est l'entité qui contient la plus petite version de la donnée encodée.
3. Le décodeur: Dans cette partie, le model apprend à reconstruire la donnée originelle en se basant uniquement sur la donnée contenue dans le goulot.
4. La perte d'information: Qui est la méthode qui mesure la qualitée de la reconstruction.

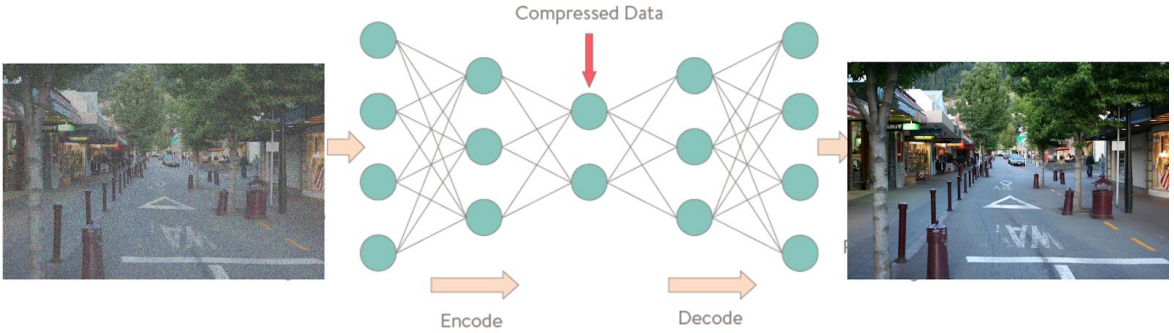


Figure 3.2.2.3 : Schéma exemple du model de l'auto encoding pour la réduction de bruit

3.2.3. Des exemples de réseau CNN :

https://towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d
https://github.com/lazyprogrammer/facial-expression-recognition
https://github.com/amineHorseman/facial-expression-recognition-using-cnn

https://github.com/d-acharya/CovPoolFER
https://github.com/neopenx/Facial-Expression

4. Approche Proposée

4.1 Architecture de notre application du système de reconnaissance d'émotions :

Nous avons prévu de faire application mobile qui récupère une image depuis l'appareil photo de l'utilisateur, les envoie à un algorithme de prétraitement. En effet, l'image retournée respectera les propriétés nécessaires à son bon traitement par la suite.
L'image passera ensuite dans un réseau de neurones CNN qui retournera l'émotion qui a été prédite suite à l'analyse de l'image.

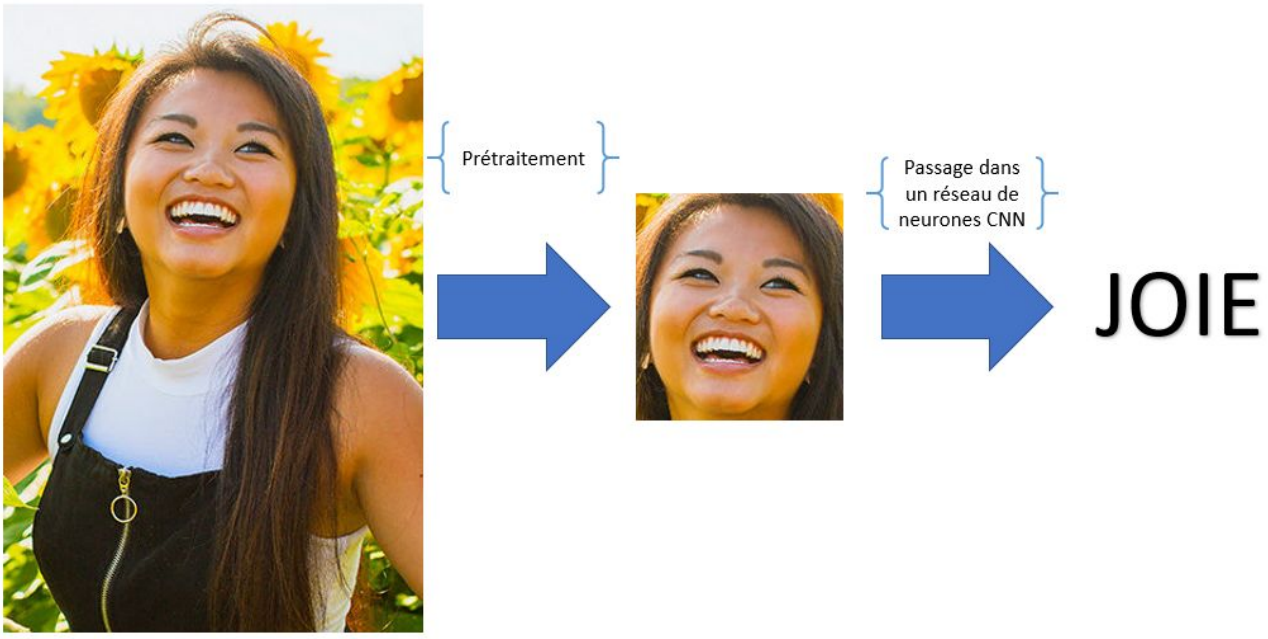


Figure 4.1.1 : Schéma simplifié du prototype de notre application

5. Conclusion

A la date du 27 juin 2020, nous avons terminé les recherches bibliographiques.
Nous savons que nous allons utiliser un réseau de neurones CNN pour traiter l'image.Il nous faut maintenant tester les différentes méthodes afin de trouver laquelle est la plus adaptée à notre format de donnée. Il nous faut ensuite mettre en place l'algorithme de prétraitement que nous construirons soit en Python avec OpenCV (qui est une bibliothèque graphique de traitement d'image utilisable en Python), soit via Matlab.
Il nous faut construire l'application en elle-même (l'interface) avec la possibilité de recevoir des images issus de l'appareil photo.

Nous avons créer un GitHub afin de mieux nous organiser pour la mise en place et le déploiement de l'application.
Le lien : https://github.com/LucasBarrot/FER2020.

6. Références Bibliographique

émotions :

- https://tel.archives-ouvertes.fr/tel-01622639/document
- https://www.irit.fr/publis/TCI/Dalle/these_mercier.pdf
- https://edutice.archives-ouvertes.fr/edutice-00000702/document
- P. Ekman. Are there basic emotions? Psychological Review, 99 :550–553, 1992
- J. A. Russell. A circumplex model of affect. Journal of personality and social psychology, 39 :1161–1178, 1980.

méthodes :

SVM :

- https://dumas.ccsd.cnrs.fr/dumas-00745988/document
- http://biblio.univ-annaba.dz/wp-content/uploads/2016/09/These-Benmohamed-Abderrahim.pdf
- https://github.com/amineHorseman/facial-expression-recognition-svm

CNN vs SVM :

- https://www.quora.com/What-is-the-difference-between-CNN-and-a-support-vector-machine#:~:text=Convolution%20Neural%20Network%20is%20non,model%20complexity%20isn't%20possible.
- https://iopscience.iop.org/article/10.1088/1755-1315/357/1/012035/pdf
- https://www.quora.com/Why-does-the-convolutional-neural-network-have-higher-accuracy-precision-and-recall-rather-than-other-methods-like-SVM-KNN-and-Random-Forest
- https://www.quora.com/Why-is-CNN-better-than-SVM
- https://www.quora.com/What-is-the-difference-between-CNN-and-a-support-vector-machine#:~:text=Convolution%20Neural%20Network%20is%20non,model%20complexity%20isn't%20possible.

Différentes méthodes :

- https://datakeen.co/3-deep-learning-architectures-explained-in-human-language/

RNN :

- https://blog.octo.com/les-reseaux-de-neurones-recurrents-des-rnn-simples-aux-lstm/
- https://penseeartificielle.fr/comprendre-lstm-gru-fonctionnement-schema/#:~:text=Un%20r%C3%A9seau%20de%20neurones%20r%C3%A9currents,tr%C3%A8s%20r%C3%A9pandu%20en%20deep%20learning.&text=Toutes%20les%20valeurs%20entrantes%20(et,de%20sortie%20pour%20chaque%20neurone

CNN :

- https://fr.wikipedia.org/wiki/R%C3%A9seau_neuronal_convolutif#:~:text=En%20apprentissage%20automatique%2C%20un%20r%C3%A9seau,est%20inspir%C3%A9%20par%20le%20cortex
- https://www.researchgate.net/figure/Schematic-diagram-of-a-basic-convolutional-neural-network-CNN-architecture-26_fig1_336805909

Auto encoding :

- https://towardsdatascience.com/auto-encoder-what-is-it-and-what-is-it-used-for-part-1-3e5c6f017726

- https://fr.wikipedia.org/wiki/Auto-encodeur

Application mobile :

- https://blog.engineering.publicissapient.fr/2017/07/24/on-device-intelligence-integrez-du-deep-learning-sur-vos-smartphones/