UQAC

Université du Québec
à Chicoutimi

Nom : BOURREAU

Prénom : Hugo

Formation : Maîtrise en informatique (professionnel)

Module : 8INF846 - Intelligence artificielle - Automne 2021 - 01

Responsable du module : Bianca M. NAPOLEAO

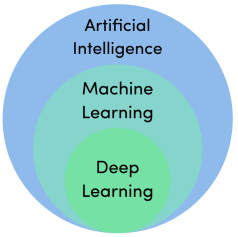
Titre du document : Analyse d'un sujet spécifique d'IA "Sign language recognition using deep learning"

Date d'envoi : Le lundi 22novembre 2021

Nombre de caractères : 7 842

☒ En adressant ce document à l'enseignant, je certifie que ce travail est le mien et que j'ai pris connaissance des règles relatives au référencement et au plagiat.

L'article choisi a comme sujet l'utilisation de l'apprentissage profond afin d'étudier la langue des signes panaméenne. L'apprentissage profond (de l'anglais deep learning) est une méthode issue de l'apprentissage automatique (machine learning). Cette intelligence artificielle va prendre un ensemble de données et de règles en entrée et construire un réseau de neurones artificiels. Cette intelligence pourra par la suite s'améliorer sans intervention humaine grâce à un apprentissage empirique.



Pour créer ce réseau de neurones, nous avons besoin de données d'entrée. Pour obtenir ces données, il existe deux types de solutions. Les systèmes basés sur les contacts, par exemple des gants avec des capteurs, malheureusement ces méthodes sont souvent dispendieuses. Il existe un autre type de système, bien plus abordable utilisant des caméras. Nous vous présenterons par la suite la solution au problème énoncé dans l'article en quatre parties: le système de vision grâce aux caméras, l'acquisition des données, l'algorithme et les résultats, puis des améliorations possibles.

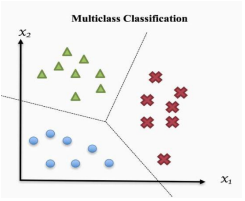
Afin de pouvoir exploiter des images, il est nécessaire d'utiliser plusieurs caméras possédant des angles de vue différents afin de pouvoir distinguer les signes. En utilisant un seul point de vue, nous perdons d'une part la notion de profondeur, et d'autre part, certains signent peuvent se ressembler et se confondre. Le système permettant de recueillir des images sera conçu avec une caméra au niveau du front ainsi qu'une seconde au niveau du torse.

Les images obtenues devront permettre de reconnaître les vingt-quatre signes de l'alphabet manuel panaméen.

Pour obtenir des données d'entrée de l'algorithme d'apprentissage profond, il a été demandé à quatre personnes de porter le système de vision et de faire chaque signe pendant dix secondes. Pour améliorer la reconnaissance, il a été demandé de faire les signes devant des paysages différents (en intérieur et en extérieur). L'objectif étant de faire varier les fonds. Les signes ont aussi été faits en utilisant un fond vert. Il a donc été possible de générer des données additionnelles en faisant varier les arrières plans des images capturées dans cette configuration. Vous pouvez le voir sur l'image ci-dessous, une image d'entrée a permis de générer plus de données pour l'algorithme sans prendre trop de temps.

Un second procédé a été utilisé pour améliorer la reconnaissance: effectuer des modifications sur une image. Par exemple déformer l'image, faire varier l'intensité lumineuse ou légèrement l'angle du signe effectué.

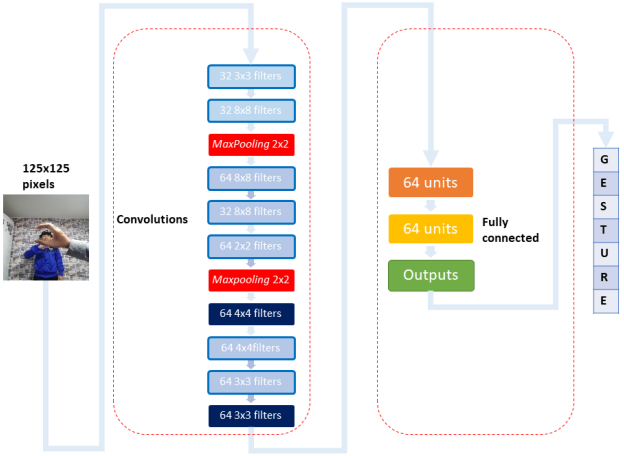
Une fois les images d'entrée récupérées, il a été nécessaire de modéliser la situation. Le problème a été vu comme multi-classes. Celui-ci a été séparé en deux sous-problèmes.



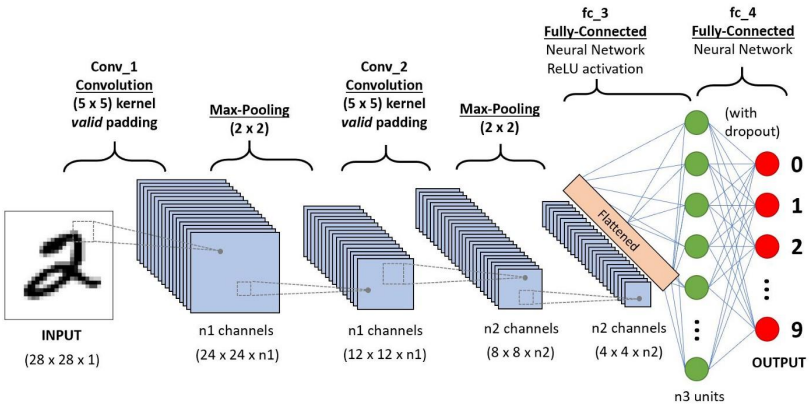
La caméra inférieure (celle au niveau du torse) permet de classifier directement un geste s'il est reconnu et qu'il n'y a pas de confusion possible. En revanche, s'il a ambiguïté, la caméra du dessus est utilisée. Les deux problèmes multi-classes sont donc comme ceci:

Model	Top View	Bottom View
Classes	B, C, D, E, F, G, H, I, L, O, P, Q, R, T, U, W, X	A, K, M, N, S, V, Y
Total	17	7

Une fois le problème modélisé, il a été possible de mettre en place l'algorithme utilisant un réseau neuronal convolutif afin de reconnaître les images. L'architecture est la suivante:



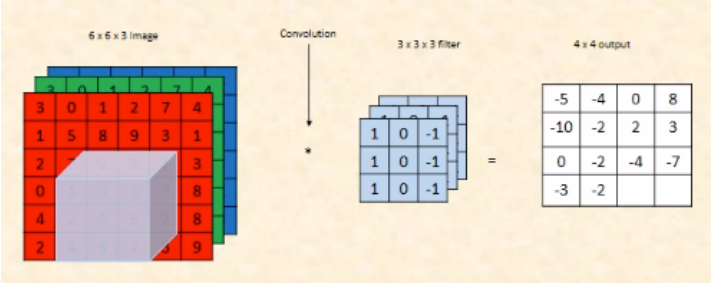
On remarque tout d'abord que les images perçues par la caméra ne sont pas directement envoyées à l'algorithme, elles subissent tout d'abord une baisse de résolution. Elles passent de 640x480p à 125x125p. Une fois le traitement effectué, les images sont données à l'algorithme permettant d'établir son modèle. Nous pouvons percevoir le modèle comme ceci:



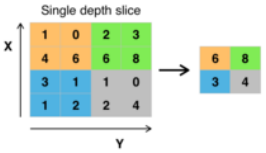
Il y a deux étapes clés dans l'algorithme pour traiter l'image et la transformer en signe reconnu: d'une part des convolutions, utilisant deux procédés que nous détaillerons par la suite, et d'autre part, un réseau de deux fois 64 neurones tous connectés reliés à la sortie. Ceux-ci seront ajustés au fil du temps permettant une amélioration de la reconnaissance. Si une image d'entrée ne donne pas la bonne sortie lors de la phase

d'entraînement, alors le réseau de neurones va s'ajuster jusqu'à obtenir une identification correcte.

S'agissant de la phase convolution, il y a deux types de traitements sur l'image. L'application de filtres, ce qui correspond à des convolutions. Il s'agit d'une opération mathématique entre deux fonctions qui vont en produire une troisième en sortie. Nous pouvons le voir comme ceci (avec des filtres 3x3x3):



L'autre traitement est le "pooling". Ici un pooling 2x2 est utilisé. A chaque étape, il y a la compression de l'image d'un facteur quatre. Chaque carré de 2x2 pixels en entrée va correspondre à un seul pixel en sortie. La réduction de l'image va permettre un gain de performance.



Un point important évoqué auparavant est la notion de phase d'entraînement. Lorsque nous donnons à l'algorithme les images récupérées, celui-ci ne donnera pas forcément les bonnes réponses au premier essai. On dira à l'algorithme s'il donne la bonne réponse ou non. C'est la phase d'entraînement. Une fois que cette étape est terminée, il peut être utilisé. Cependant l'algorithme n'est pas gravé dans le marbre, celui-ci s'ajustera toujours au fil du temps, améliorant son efficacité.

Vous pouvez observer les résultats obtenus sur la droite après une utilisation normale de l'algorithme. Pour chaque lettre, il y a deux résultats, la précision (precision) et le rappel (recall). Les notations suivantes seront utilisées pour l'explication:

True Positive (TP)	False Positive (FP)
False Negative (FN)	True Negative (TN)

Le rappel est calculé comme le nombre de fois où le signe est reconnu sur le nombre de fois où il est apparu:

$$\text{Recall} = \frac{TP}{TP + FN}$$

Tandis que la précision est le nombre de fois où l'élément a été correctement identifié sur le nombre de fois où l'algorithme pense l'avoir identifié:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Bottom view model

	Precision	Recall
A	0,79	1,00
K	0,48	0,73
M	0,54	1,00
N	0,28	0,53
S	0,00	0,00
V	1,00	0,27
Y	0,00	0,00

Top view model

	Precision	Recall
B	0,69	0,73
C	0,39	1,00
D	0,50	0,73
E	0,52	1,00
F	1,00	0,65
G	0,65	1,00
H	1,00	0,67
I	0,67	0,65
L	0,65	1,00
O	0,67	0,93
P	0,00	0,00
Q	1,00	1,00
R	1,00	0,80
T	1,00	1,00
U	1,00	0,60
W	0,88	1,00
X	0,00	0,00

Pour illustrer le rappel et la précision, on peut prendre la lettre R. La précision est de 1.00, ce qui indique que l'ensemble des R identifiés étaient corrects. Cependant le rappel est à 0.80, ainsi parmi l'ensemble des R qui sont apparus, un signe sur 5 n'a pas été reconnu.

En étudiant l'ensemble des résultats, on remarque qu'ils ne sont pas parfaits. Certains signes ont des difficultés à être reconnus, comme les lettres C, D, E ou K. Cependant, à la lecture d'un mot, si une lettre est mal identifiée, le message pourrait toujours être compréhensible. Dans le cas où plusieurs lettres seraient erronées, le taux de compréhension pourrait varier…

Le problème d'identification de la langue des signes est complexe, de part le nombre de signes possibles et l'expression de ceux-ci qui ne sont jamais exactement les mêmes (position, inclinaison…). Afin d'améliorer les résultats, il faudrait reprendre la solution proposée et trouver différents moyens d'améliorer le système.

L'article étudié a été publié le 1er Août 2020 et est en anglais. Pour en savoir plus, l'auteur a aussi ajouté le lien de la thèse (en Espagnol) réalisée sur le même sujet et parue en 2019. J'ai donc contacté l'auteur sur LinkedIn en lui demandant si, avec le recul, il avait des idées d'amélioration ou de nouvelles méthodes plus efficaces pour répondre au problème. Il m'a expliqué que, dans ses travaux, il a modélisé le problème comme de la détection d'image et que c'est pour cette raison qu'il utilise un réseau de neurones. Cependant, utiliser de la détection d'objet avec d'autres algorithmes (comme Fast RCNNs, RetinaNet, YOLO, …) permettrait d'améliorer les performances. Selon lui, la solution serait d'utiliser un autre type d'algorithme, de changer la méthode utilisée.

Pour finir, je lui ai demandé si utiliser une caméra et se déplacer autour de la personne afin d'obtenir les visions sous d'avantages d'angles pourrait être une bonne idée. Il m'a confirmé que ce serait intéressant car au lieu de générer des données artificiellement (avec des distorsions ou en changeant les fonds), il y aurait d'avantages de données réelles collectées.

BIBLIOGRAPHIE

La source de l'article utilisé est la suivante:

Herazo, J. (1er Août 2020). Sign language recognition using deep learning : *A dual-cam first-person vision translation system*. Towards Data Science, repéré à https://towardsdatascience.com/sign-language-recognition-using-deep-learning-6549268c60bd (dernière consultation le 18/11/2021 - 2H38 PM EST)

Vous pourrez retrouver sur la page le lien d'une vidéo youtube de démonstration de l'algorithme ainsi que la thèse de l'auteur sur le sujet (en Espagnol) :

Herazo, J. (2 Août 2020). *Sign language recognition with convolutional neural networks* [Fichier vidéo]. Repéré à https://youtu.be/pGZ7fPvnU6Q

Herazo, J. (2 Juin 2020). Reconocimiento de señas de la lengua de señas panameña mediante aprendizaje profundo. GitHub, repéré à https://github.com/joseherazo04/SLR-CNN/blob/master/Jos%C3%A9%20Herazo%20TFM.pdf

Afin de comprendre le fonctionnement de l'intelligence artificielle et du deep learning, plusieurs sites ont été utilisés:

- https://en.wikipedia.org/wiki/Convolution (fonctionnement des convolutions)
- https://fr.wikipedia.org/wiki/R%C3%A9seau_neuronal_convolutif (réseaux de neurones)
- https://gaussian37.github.io/dl-concept-cnn/
- https://fr.wikipedia.org/wiki/Pr%C3%A9cision_et_rappel
- https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall?hl=fr