Polytechnique Montréal

INF8225

Intelligence artificielle : techniques probabilistes et d'apprentissage

Apprentissage de différentes tâches avec un réseau de neurones convolutionnel

Auteurs :
Laurent Desmet (1865564)
Paul Margheritta (1800303)
Pierre Zins (1863527)

18 avril 2017

Table des matières

1 Présentation du problème **2**
1.1 Objectifs . . . 2
1.2 Travaux antérieurs . . . 2
1.3 Techniques employées . . . 3
1.4 Cheminement suivi . . . 3

2 Préparation des données **4**
2.1 Caractéristiques de l'ensemble de données . . . 4
2.2 Pré-traitement . . . 4
2.3 Augmentation de l'ensemble de données . . . 5

3 Expériences réalisées **6**
3.1 Classification selon le genre . . . 6
3.2 Classification selon l'âge . . . 8
3.3 Régression selon l'âge . . . 11
3.4 Déterminations séquentielle et parallèle du genre et de l'âge . . . 14

4 Analyse critique **16**

5 Conclusion **17**

1 Présentation du problème

1.1 Objectifs

L'objectif principal de ce projet est, étant donné la photographie d'une personne humaine, de déterminer automatiquement son genre (homme ou femme) et une approximation de son âge (en années). Une telle estimation peut avoir de nombreuses applications, par exemple dans le cadre d'une enquête policière, à des fins de statistiques ou de marketing, ou encore pour construire des interfaces utilisateur avancées et intelligentes.

La tâche à réaliser est complexe car elle n'est déjà pas évidente pour une personne humaine. En effet, s'il est dans la plupart des cas relativement aisé pour un humain de donner le genre d'une personne à partir d'une photo, déterminer un âge peut en revanche conduire à une erreur importante. De nombreux facteurs peuvent influencer négativement la qualité de la détermination de l'âge voire du genre : le physique de la personne, le maquillage, la position et l'orientation de son visage sur la photo, l'éclairage, etc. Il n'est donc pas étonnant que l'estimation de l'âge par une personne humaine engendre une erreur absolue (rajeunissement ou vieillissement) d'une dizaine d'années.

Ainsi, la détermination automatique du genre et de l'âge à partir d'une photo risque de présenter un certain nombre d'obstacles. Notre objectif sera donc, pour un ensemble d'images, d'obtenir un maximum d'estimations exactes du genre et de l'âge, le tout en un temps raisonnable et moyennant une consommation de ressources acceptable.

1.2 Travaux antérieurs

De nombreux articles s'intéressent au problème d'estimation du genre ainsi que de l'âge d'une personne à partir d'une photo.

Jia, Lansdall-Welfare et Cristianini [2] proposent trois architectures différentes de réseaux convolutionnels afin de faire une classification selon le genre des personnes. Leur architecture la plus profonde et la plus coûteuse en terme de ressources ressemble à un VGG16 [6] et donne les meilleurs résultats (98,9 % de précision sur l'ensemble de test). La base d'images utilisée est LFW (*Labeled Faces in the Wild*) [1]. Les auteurs montrent également l'intérêt de conserver une certaine marge autour du visage afin d'améliorer les résultats. Pour justifier cela, ils expliquent que les humains sont capables d'identifier le genre d'un personne en voyant une bordure autour du visage d'une personne, sans voir le visage. Ainsi des informations importantes sont présentes dans cette bordure, comme la forme du visage ou encore les cheveux.

Antipov, Berrani et Dugelay [1] proposent une architecture de base plus légère avec plusieurs optimisations. Avec la même base d'images, ils obtiennent des résultats proches de Jia, Lansdall-Welfare et Cristianini [2] mais avec un temps d'apprentissage réduit.

Levi et Hassner [3] s'appuient sur un réseaux convolutionnel composé de trois couches de convolution et deux couches denses. Le point essentiel de leur travail concerne la base d'images utilisée : Adience [2]. Cette dernière est constituée de photos récupérées directement depuis des smartphones. Les auteurs expliquent que ces images reflètent mieux des photos réelles par rapport à d'autres ensembles où les images sont très contraintes ou énormément pré-traitées. Adience présente des photos avec une grande variation au niveau de la position de la tête, des conditions d'éclairage ou encore de la qualité de la photo. Malgré cela, les auteurs obtiennent des résultats satisfaisants, leur meilleure méthode atteint une précision de 86,8 %. Levi et Hassner [3] ont également utilisé

1. `http://vis-www.cs.umass.edu/lfw/`
2. `http://www.openu.ac.il/home/hassner/Adience/data.html#agegender`

leur architecture de réseau convolutionnel à cinq couches pour estimer l'âge des personnes sur les photos.

Enfin, Malli, Aygün et Ekenel [5] proposent une architecture avec trois réseaux convolutionnels VGG16 [6] utilisés en parallèle. Leur méthode sera rapidement présentée dans la suite.

1.3 Techniques employées

En rapport avec les techniques abordées dans le cadre du cours, nous choisissons d'exploiter l'idée d'apprentissage supervisé. En effet nous disposons de plusieurs bases de données de visages étiquetées avec le genre et/ou l'âge de la personne, telles que LFW ou IMDB-WIKI [3] (du nom des sites web Internet Movie Database et Wikipédia). Afin de réaliser notre apprentissage, nous nous basons sur un sous-ensemble de la base IMDB-WIKI qui présente plus de 60 000 images avec des étiquettes facilement accessibles et exploitables.

Pour l'apprentissage en tant que tel, nous utilisons un réseau de neurones convolutionnel (*Convolutional Neural Network*, CNN). Ces réseaux représentent en effet un choix classique et efficace pour les applications liées à la reconnaissance d'images, en raison de leur structure exploitant des opérations de convolution bien adaptées à ce type de traitement. Nous nous concentrerons sur la recherche de l'architecture optimale et des meilleurs hyperparamètres pour chacune de nos expériences.

L'implémentation de l'apprentissage par réseau de neurones convolutionnel est réalisée en Python avec la librairie Keras [4] qui offre une interface de haut niveau simple, complète et efficace pour l'apprentissage machine avec Theano ou TensorFlow. TFLearn [5] est également utilisé pour générer facilement des graphes à l'aide de TensorBoard [6]. Les calculs sont accélérés par processeur graphique (GPU Nvidia GTX 950M).

1.4 Cheminement suivi

Les images de la base de données devront dans un premier temps subir un pré-traitement dans le but de sélectionner uniquement celles appropriées pour l'apprentissage et de les optimiser à cette fin. L'ensemble de données pourra également être augmenté par le biais de transformations (rotations, décalages, zooms, etc.).

La première étape importante de l'estimation du genre et de l'âge consiste à tenter de déterminer ces valeurs séparément, chacun avec un réseau convolutionnel distinct. La détermination du genre se résume à une classification. Celle de l'âge peut prendre la forme d'une classification dans un certain nombre de classes d'âge, mais surtout d'une régression permettant d'obtenir une estimation directe de l'âge.

Une étape plus avancée consiste à analyser les interactions possibles entre les deux apprentissages. Connaissant déjà l'information du genre, il est en effet possible que la détermination de l'âge soit facilitée. Nous pourrions donc imaginer d'estimer les deux valeurs avec le même réseau convolutionnel séquentiellement. Une autre idée pour améliorer les résultats de l'apprentissage consiste à séparer en deux le réseau de neurones au moment d'entrer dans la composante complètement connectée, pour tenter d'apprendre parallèlement les données de genre et d'âge.

3. `https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/`
4. `https://keras.io/`
5. `http://tflearn.org/`
6. `https://www.tensorflow.org/get_started/summaries_and_tensorboard`

2 Préparation des données

2.1 Caractéristiques de l'ensemble de données

L'ensemble de données que l'on utilise a une réelle importance car c'est sa qualité qui va être déterminante sur la performance de l'entraînement. Nous veillons à trouver un ensemble de données suffisamment vaste pour traiter tous les types d'entrées auxquels notre algorithme pourrait être confronté.

De plus, notre base de données ne contient que des images dont l'âge est supérieur à 10 ans et inférieur à 100 ans. Néanmoins, nous verrons plus tard que la distribution des âges suit une approximation de la loi du χ_2 ce qui a l'inconvénient de rendre une partie de la population mieux entraînée qu'une autre. Toutes les images fournies et étiquetées par l'ensemble de données ne sont pas parfaites. Certains visages présentent des occlusions (lunettes de soleil, chapeau...), d'autres sont en noir et blanc, et enfin elles n'ont pas toutes la même taille. Certaines images sont mêmes des voitures ou autres objets. Afin de veiller à ne pas entraîner notre modèle sur des cas particuliers, ce qui empêcherait notre algorithme de bien généraliser, nous allons avoir recours à l'utilisation d'un pré-traitement. Ce dernier sera effectué à l'aide de la librairie OpenCV en Python.

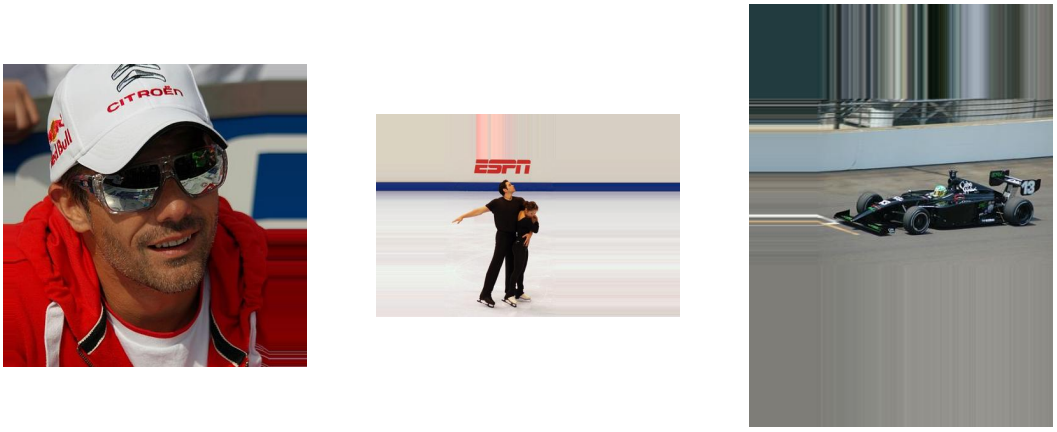

FIGURE 1 – Exemples d'images non pertinentes pour notre algorithme

2.2 Pré-traitement

La première chose à faire et donc de correctement sélectionner nos données pour pouvoir entraîner notre algorithme. Ainsi nous excluons d'office toutes les images dont le genre est « indéterminé », c'est-à-dire non reconnu comme humain dans l'ensemble de données. De plus, pour prévenir toute erreur d'étiquetage sur l'âge, on exclut ceux ayant un âge négatif ou supérieur à 100. Enfin, nous gardons seulement les images en couleur car nous avons constaté que l'utilisation d'un seul canal génère des résultats moins satisfaisants, ce qui concorde avec les articles que nous avons lus.

Une fois que ce travail est réalisé, on utilise un détecteur de visage fourni par la librairie OpenCV. Cet algorithme permet à partir d'une image de déterminer s'il y a un visage et d'en sortir une *bounding box* centrée sur le visage. Parmi les différentes versions existantes, nous avons utilisé celles basées sur la décomposition en ondelettes de Haar et sur un descripteur de texture LBP. Les deux résultent en l'utilisation des mêmes images à quelques exceptions près. Ainsi, certaines images ne passent pas le test (c'est le cas de la première image ci-dessus par exemple) car l'occlusion est trop importante dans ce cas. Si l'image passe le test, on obtient un résultat proche de ceux ci-dessous :

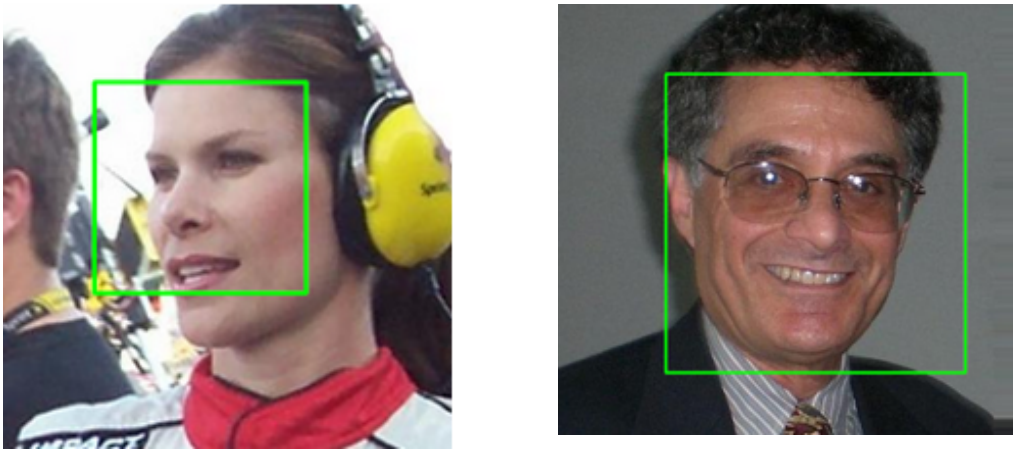

FIGURE 2 – Exemples d'images avec la détection du visage

Néanmoins, comme on peut le voir avec l'exemple ci-dessus, cette technique centre le visage et occulte les cheveux et parfois le menton. Or, ces deux parties peuvent être parfois importante dans la régression de l'âge et même la détection du genre. Ainsi, une personne chauve avec de la barbe a objectivement plus de chance d'être un homme de 40 ou 50 ans qu'une femme de 25 ans. On va alors augmenter automatiquement la taille de nos *bounding boxes* de 50 % en hauteur et en largeur. On verra dans la section *Expériences réalisées* l'effet de l'utilisation de cette marge.

Au final, une fois cet algorithme utilisé on redimensionne notre image par interpolation cubique pour réduire la taille de l'image. En effet, il faut que toutes les images soient de même taille, et si possible de résolution assez faible car nous sommes limités par la mémoire de nos GPU. Généralement, nous avons utilisé des images de taille 32×32 pour le genre et 64×64 ou 128×128 pour l'âge.

2.3 Augmentation de l'ensemble de données

Après avoir effectué ces étapes, nous observons une baisse notable du nombre d'images exploitables. Ainsi, sur les 60 000 images offertes par l'ensemble de données, seulement la moitié passe avec succès le pré-traitement. Cette quantité nous semble relativement faible. C'est pourquoi nous allons effectuer des transformations géométriques simples pour pouvoir doubler la taille de l'ensemble de données. Ainsi, on choisit aléatoirement de faire une translation et/ou une rotation sur chaque image de l'ensemble de données. L'utilisation de ces transformations résulte, dans la plupart de nos expériences, en une légère augmentation des performances.

FIGURE 3 – Exemples d'images après transformation

Au final, notre ensemble de données se compose de 60 000 images en couleur de même format, centrées sur le visage avec utilisation d'une marge supplémentaire pour pouvoir fournir à l'algorithme d'apprentissage le plus de détails possibles.

3 Expériences réalisées

3.1 Classification selon le genre

La première expérience consiste à classifier les entrées dans une des deux catégories suivantes : homme ou femme. Le genre est codé dans les étiquettes sous la forme de vecteurs *one-hot* de taille 2. L'architecture choisie pour le réseau convolutionnel comporte quatre convolutions.

Inputs : $32 \times 32 \times 3$
Conv : 32 @ 3×3 (ReLU)
Conv : 32 @ 3×3 (ReLU)
MaxPool : 2×2
Conv : 64 @ 3×3 (ReLU)
Conv : 64 @ 3×3 (ReLU)
MaxPool : 2×2
MLP : 256 (ReLU)
MLP : 256 (ReLU)

FIGURE 4 – Architecture du réseau convolutionnel pour la classification selon le genre

L'entraînement est réalisé sur 70 % de l'ensemble de données, avec 15 % d'autres entrées utilisées pour l'ensemble de validation. Les paramètres suivants ont été fixés :

- **taux de *dropout*** : 0,25 pour la partie convolutionnelle, 0,5 pour la partie complètement connectée ;
- **initialisation des poids** : initialisation normale de He (`he_normal`, permet une convergence rapide) ;
- **fonction d'activation de la sortie** : sigmoïde (`sigmoid`, utile pour une classification binaire) ;
- **fonction de perte** : entropie croisée catégorielle (`categorical_crossentropy`) ;
- **rétropropagation** : descente du gradient stochastique avec optimisation RMSProp (`rmsprop`).

Nous avons visualisé la précision ainsi que la perte sur l'ensemble de validation durant l'apprentissage. Nous avons pu comparer les versions avec et sans marge supplémentaire appliquée sur les visages. On peut voir que le fait de conserver une marge autour du visage aide l'apprentissage, principalement au niveau de la vitesse de l'apprentissage. De plus, la précision finale obtenue sur l'ensemble de test est meilleure.

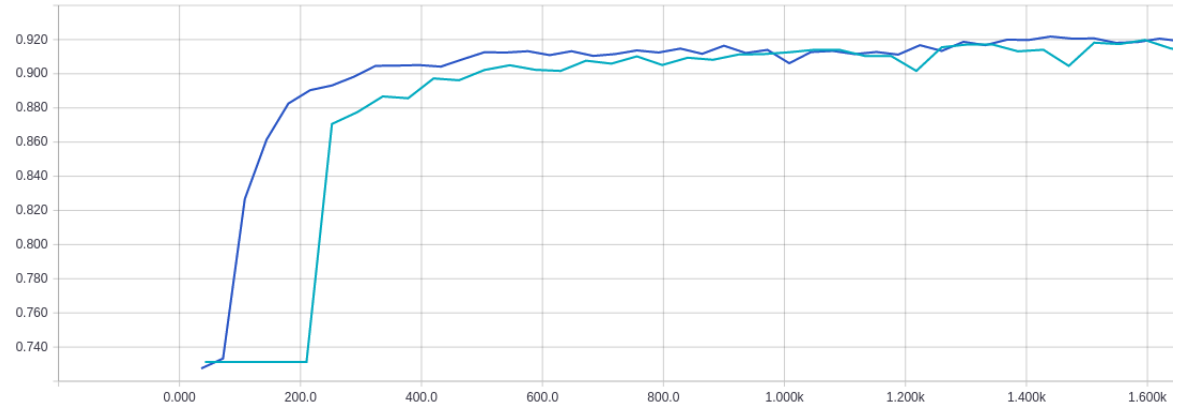

FIGURE 5 – Précision sur l'ensemble de validation. **bleu** : avec marge, **vert** : sans marge

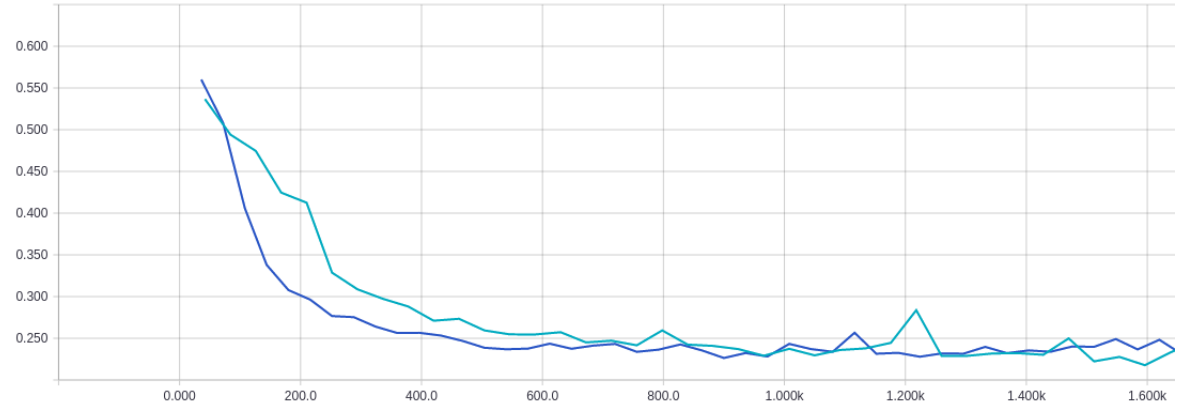

FIGURE 6 – Fonction de perte sur l'ensemble de validation. **bleu** : avec marge, **vert** : sans marge

À l'issue de l'apprentissage, nous obtenons une matrice présentant, pour chaque entrée, la probabilité d'appartenance à chacune des deux classes. Ces prédictions sont comparées aux étiquettes afin de déterminer la précision sur l'ensemble de test (choisi pour représenter les derniers 15 % de l'ensemble de données). Nous obtenons une précision de 92,0 % sur l'ensemble de test. La précision finale sur l'ensemble d'entraînement s'élève à 98,9 %.

La précision obtenue sur l'ensemble de test est moyennement satisfaisante. Elle est certes plutôt élevée (plus de 9 visages sur 10 sont correctement classifiés) mais elle reste peu intéressante par rapport à la précision du jugement humain, que l'on peut deviner supérieure à 99 %. Une marge d'amélioration importante existe donc encore dans la classification de selon le genre. Les résultats pourraient sûrement être améliorés en utilisant un réseau convolutionnel plus complexe, mais nous sommes limités en termes de ressources, principalement en mémoire GPU.

Une autre idée pouvant peut-être améliorer les résultats serait de modifier légèrement la base d'images utilisée. Cette dernière contient environ 73% d'hommes contre seulement 27% de femmes. Afin d'égaliser la proportion, nous pourrions faire l'augmentation des données uniquement sur les images de femmes. Finalement, nos résultats sont malgré tout élevés par rapport à certains résultats présentés dans des articles, comme Levi et Hassner [3]. Cependant, il faut bien prendre en compte la base d'images utilisée. Les personnes présentes dans notre base d'images sont toutes âgées d'environ 10 ans ou plus. Nous ne rencontrons donc pas le problème d'estimation du genre sur des enfants très jeunes, pour lesquels la tâche est souvent bien plus compliquée (même pour un humain). De plus, après notre pré-traitement, la plupart de nos images sont de bonne qualité, avec un éclaire ainsi qu'une orientation du visage avantageux.

3.2 Classification selon l'âge

Nous avons ensuite tenté de classifier les entrées selon le critère d'âge. Les indications de date de naissance et de date de prise de la photo dans la base de données nous ont permis, pour chaque entrée, d'attribuer une étiquette de classification d'âge sous la forme d'un vecteur *one-hot*. La taille de ce vecteur est égale au nombre de classes d'âge que nous choisissons. Il existe de nombreuses manières de distribuer les classes d'âge, avec des résultats divers comme nous le verrons un peu plus loin. L'architecture retenue pour le réseau convolutionnel comporte cette fois trois convolutions.

Inputs : $128 \times 128 \times 3$
Conv : 96 @ 7×7 (ReLU)
MaxPool : 3×3
Conv : 256 @ 5×5 (ReLU)
MaxPool : 3×3
Conv : 384 @ 3×3 (ReLU)
MaxPool : 3×3
MLP : 512 (ReLU)
MLP : 512 (ReLU)

FIGURE 7 – Architecture du réseau convolutionnel pour la classification selon l'âge

Les paramètres sont identiques au cas précédent sauf pour la fonction d'activation de la sortie : c'est cette fois `softmax` qui est utilisée.

Les précisions obtenues sont très variables selon la distribution choisie pour les classes d'âge. Il est possible d'obtenir une précision élevée sur l'ensemble de test pour une classification binaire (plus ou moins de 50 ans) mais ce type de classification n'est pas réellement dans l'esprit de ce qui était attendu et donne trop peu d'informations. Une idée naïve est de former des classes d'âge de même taille : 10 à 20 ans, 20 à 30 ans, 30 à 40 ans, etc. Une idée plus avancée consiste à constituer des classes d'âge plus finement en adaptant la taille des classes à la réalité de la perception des âges chez l'humain : par exemple, on pourrait avoir plusieurs classes pour l'intervalle de 10 à 20 ans, mais une seule pour celui de 70 à 90 ans. De cette façon, il est également possible de s'adapter en fonction de la distribution des âges de notre ensemble de données.

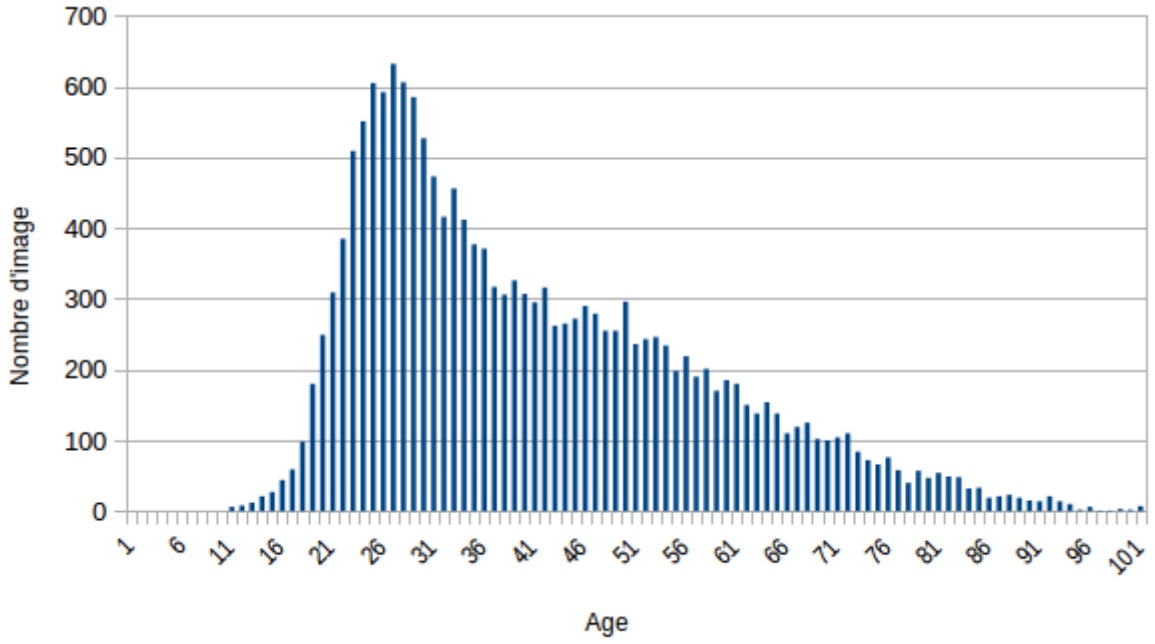

FIGURE 8 – Distribution des âges dans notre ensemble de données

Nous avons plus d'image de personnes ayant entre 20 et 30 ans. Il serait donc intéressant d'avoir des classes plus fines dans cet intervalle d'âge.

Pour évaluer notre réseau convolutionnel, nous avons utilisé les classes suivantes :

- < 20 ;
- 20-25 ;
- 30-35 ;
- 35-45 ;
- 45-60 ;
- < 60.

Ces sept classes semblaient intéressantes puisqu'elles ne sont pas trop larges, correspondent à différentes étapes de la vie et sont adaptées à la distribution des images de notre ensemble de données. Cependant, sur l'ensemble de test, nous obtenons une précision seulement de 30 %.

Voici les graphes représentant l'évolution de la précision et de la fonction de perte sur l'ensemble de validation durant la phase d'apprentissage :

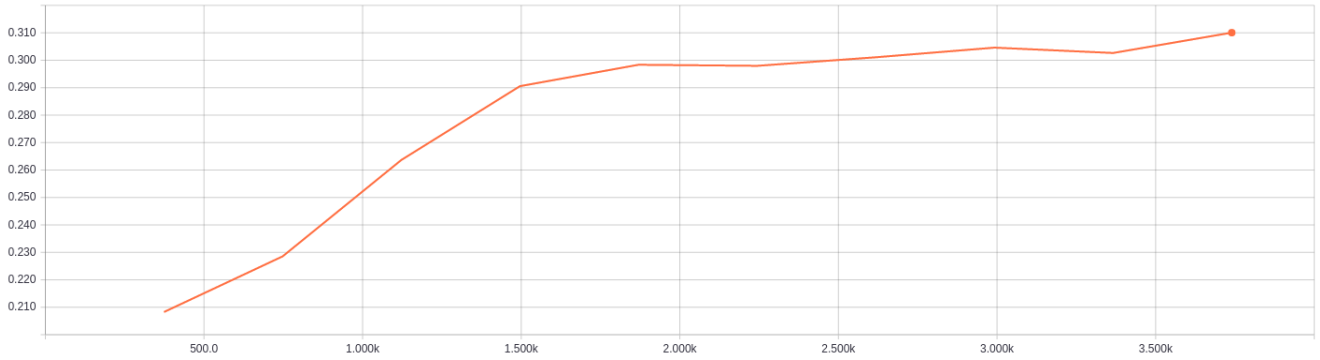

FIGURE 9 – Classification de l'âge : précision sur l'ensemble de validation

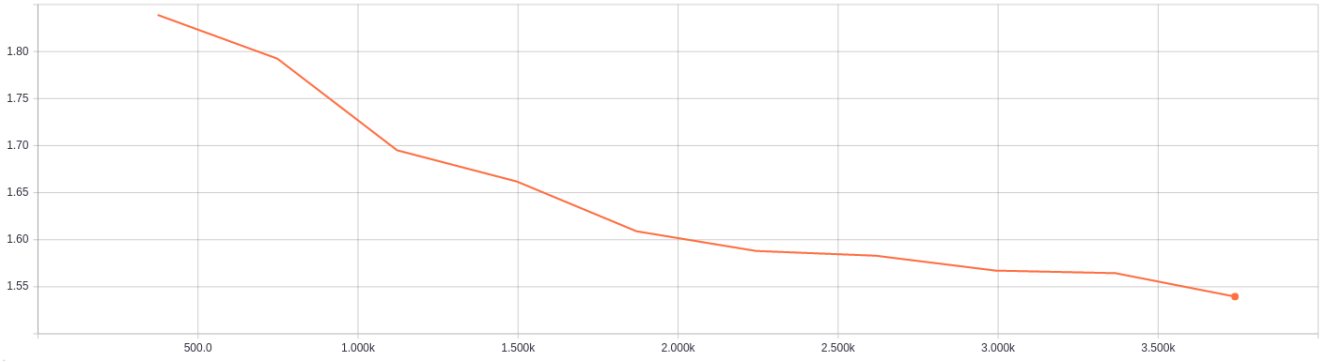

FIGURE 10 – Classification de l'âge : fonction de perte sur l'ensemble de validation

Dans l'idéal, nous aurions souhaité refaire une méthode proche de celle proposée par Malli, Aygün et Ekenel [5]. Elle consiste à combiner trois réseaux convolutionnels en parallèle et de faire une moyenne ensuite pour déterminer l'âge. De plus, chacun des trois réseaux est utilisé avec des catégories d'âges différentes. Dans les trois cas, les groupes contiennent trois années mais un décalage est appliqué.

Nous n'avons pas continué dans cette direction en raison de la faible performance de notre réseau. Dans l'article, les auteurs atteignent une précision proche de 70 %. De plus, leur classes sont bien plus petites que les nôtres. Nous obtenons une précision de seulement 30 % avec des classes plus grandes. La grande différence est probablement due au type de réseaux utilisés. Les auteurs utilisent trois VGG16 [6] en parallèle, ce qui demande énormément de ressources et que nous ne pouvons pas utiliser.

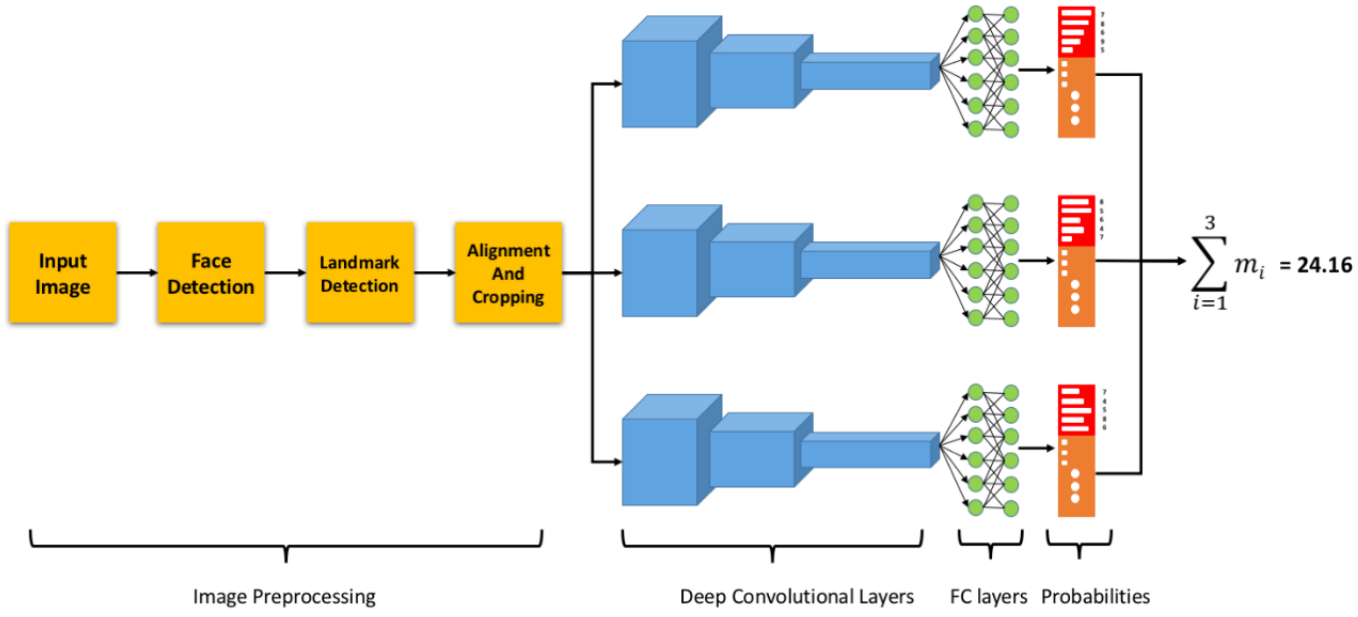

FIGURE 11 – Architecture pour la classification de l'âge

FIGURE 12 – 3 groupes avec un décalage

3.3 Régression selon l'âge

La classification selon l'âge reste imprécise et très variable selon la distribution choisie. Nous avons souhaité nous concentrer sur l'idée de régression, qui permet d'obtenir directement une estimation de l'âge pour chacune des entrées. Cette estimation est comparée avec l'étiquette (l'âge réel) pour déterminer l'erreur de l'estimation. La métrique utilisée pour quantifier le succès de l'estimation pour la régression est l'erreur absolue moyenne (*Mean Absolute Error*, MAE), qui est la moyenne des erreurs absolues entre nos prédictions et l'âge réel correspondant. L'erreur quadratique moyenne (*Mean Square Error*, MSE), est également exploitable. Le réseau proposé compte quatre convolutions.

Inputs : $32 \times 32 \times 3$
Conv : 32 @ 3×3 (ReLU)
Conv : 32 @ 3×3 (ReLU)
MaxPool : 2×2
Conv : 64 @ 3×3 (ReLU)
Conv : 64 @ 3×3 (ReLU)
MaxPool : 2×2
MLP : 2048 (ReLU)
MLP : 1024 (ReLU)

FIGURE 13 – Architecture du réseau convolutionnel pour la régression selon l'âge

La fonction d'activation de la sortie est cette fois linéaire, ce qui est un choix classique pour un problème de régression.

L'apprentissage se termine avec une erreur absolue moyenne de 6,8 ans sur l'ensemble d'entraînement. Nous obtenons une erreur absolue moyenne de 8,7 ans sur l'ensemble de test. La qualité de l'estimation présente de grandes disparités selon les exemples. Afin d'avoir plus de détails sur ces différences d'estimation, nous avons tracé l'histogramme présentant la répartition des erreurs relatives (on choisit typiquement une erreur négative pour un rajeunissement et une erreur positive pour un vieillissement). Nous observons empiriquement que cette répartition se modélise bien par une loi normale centrée en une valeur proche de 0 (ce qui est cohérent puisqu'on considère des erreurs relatives) et d'écart type proche de 12 ans (ce qui rend bien compte des fortes disparités observées). Cette modélisation permet d'estimer la probabilité selon laquelle la détermination de l'âge par régression sera exacte.

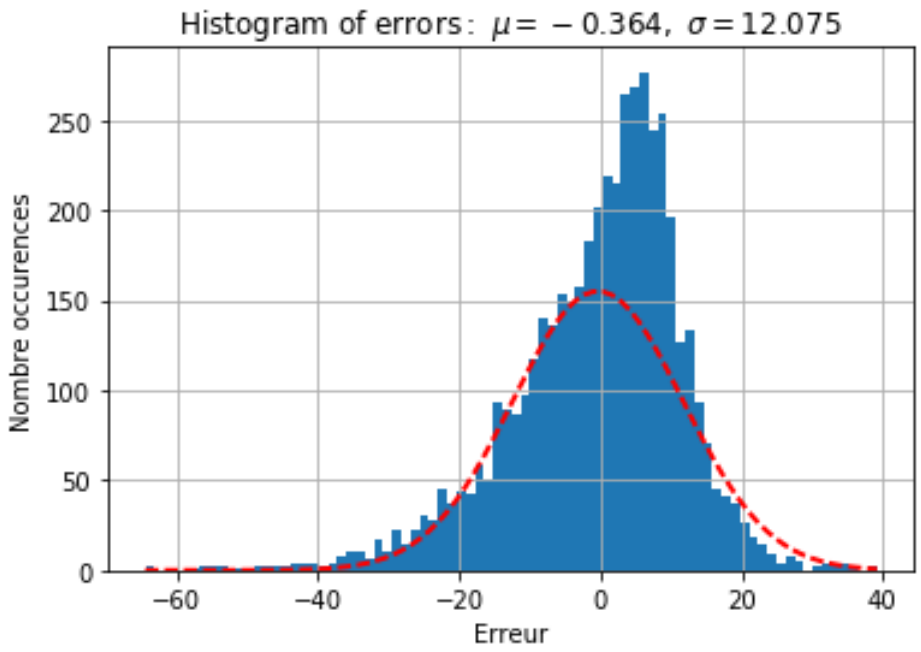

FIGURE 14 – Histogramme de la répartition des erreurs relatives pour la régression selon l'âge

Ce graphique nous permet de visualiser une distribution probabiliste de l'erreur commise sur la prédiction sous la forme d'une loi normale de paramètre (μ, σ). Ces informations sont directement exploitables et permettent de pouvoir améliorer notre algorithme considérablement. En ce sens, il suffirait de retrancher la moyenne pour améliorer nos prédictions si les résultats avaient tendance à avoir une moyenne non nulle. Si en revanche, on ne peut pas modifier certains paramètres pour réduire la variance, cela nous permet d'utiliser une nouvelle métrique pour juger la capacité de notre algorithme à être précis. En effet, grâce aux propriétés des lois normales, on peut par exemple affirmer que l'on est sûr d'estimer l'âge d'une personne à 68,27 % dans un intervalle $[-12; 12]$. Ceci semble large, mais des erreurs humaines de 6 ans et plus sont fréquentes. Ainsi un intervalle de confiance étroit est difficile à obtenir et c'est pourquoi nous souhaitons un écart type le plus proche de 0.

Les résultats de la régression sont finalement plutôt satisfaisants si l'on s'en tient aux erreurs moyennes. Nous pouvons imaginer que l'erreur absolue moyenne obtenue (moins de 10 ans) est comparable, voire inférieure, à celle qui pourrait être observée par l'estimation de l'âge par des personnes humaines. Il faut néanmoins remarquer qu'une estimation humaine ne générerait pas une telle disparité des erreurs. En particulier, on peut se douter qu'une personne humaine ne pourrait jamais se tromper avec un écart de plus de 40 ans, contrairement à notre programme d'apprentissage automatique.

Nous avons également visualisé la distribution des erreurs de notre régression en fonction des âges :

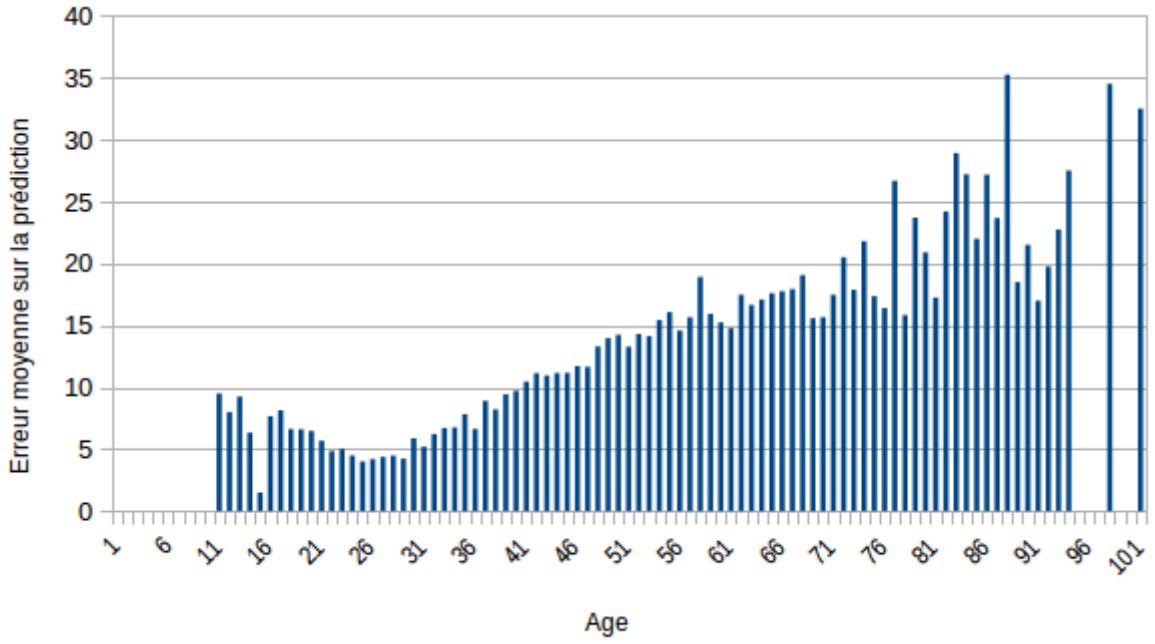

FIGURE 15 – Histogramme de l'erreur moyenne selon l'âge

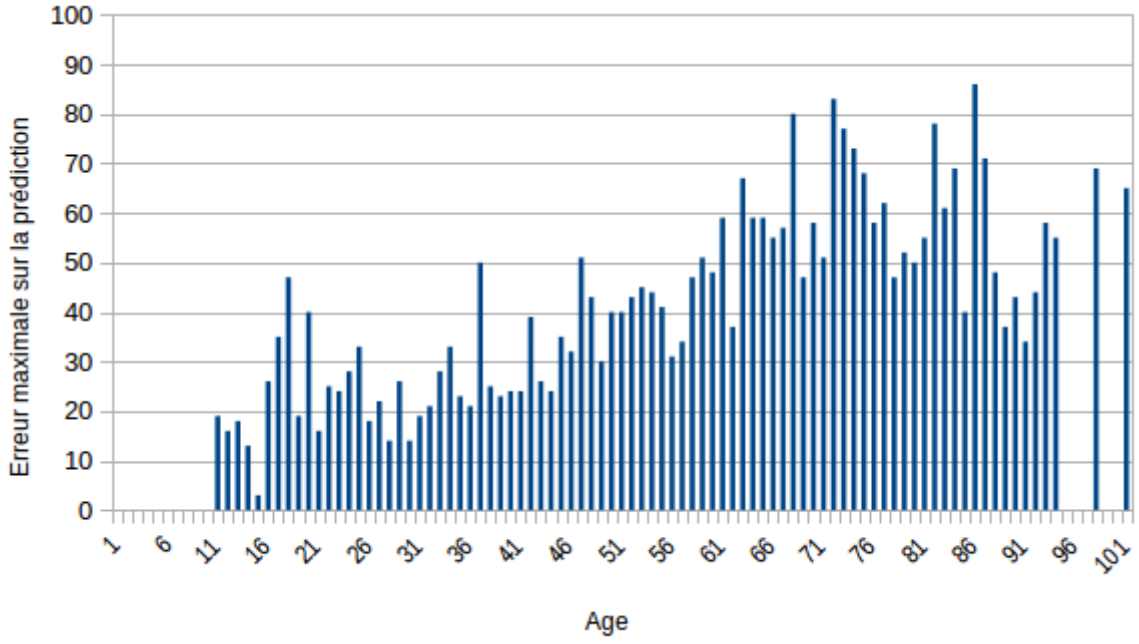

FIGURE 16 – Histogramme de l'erreur maximale selon l'âge

On remarque qu'en général, notre modèle commet plus d'erreurs (en moyenne et en maximale) pour des âges plus importants. Cela est probablement lié à la distribution des âge de la base de données d'images utilisée. En effet, les erreurs commises sont bien plus faibles pour les âges pour lesquels nous avons un nombre important de données. À nouveau, nous aurions pu corriger cela en modifiant notre base de données. Nous pourrions faire l'augmentation des données uniquement ou de manière plus importante pour certaines classes d'âge (celles contenant le moins d'images). Pour preuve, on trouvera ci-dessous le même histogramme qu'à la figure 14 mais avec seulement une partie de la population comprise entre 20 ans et 40 ans. Nous avons donc de nouveau lancé

une expérience dans la partie de la population où les âges étaient les plus fréquents. Comme attendu, dans notre cas, l'erreur quadratique moyenne sur l'ensemble de test tombe à 3,54 ans et nous obtenons l'histogramme des erreurs relatives suivant :

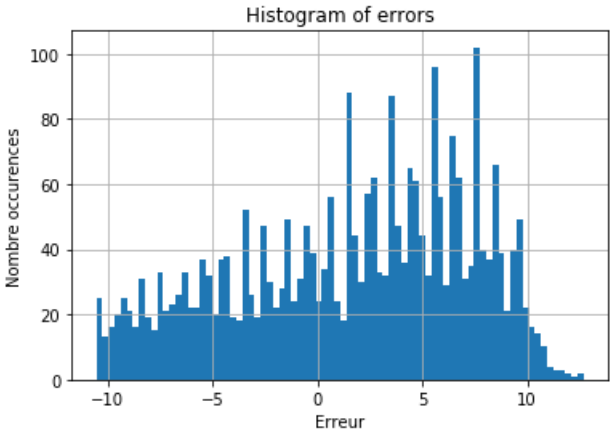

FIGURE 17 – Histogramme de la répartition des erreurs relatives pour la régression selon l'âge pour une population comprise entre 20 et 40 ans

Ce graphique peut s'apparenter à une distribution uniforme des erreurs centrée en 0. En considérant le même pourcentage de confiance que précédemment, on peut envisager non plus un écart de ± 12 ans mais de ± 7. Cela reste à relativiser car la distribution n'est pas parfaitement uniforme, mais cela confirme bien qu'avec un meilleur ensemble d'images, nous pourrions augmenter les performances du réseau. Remarquons de plus qu'il ne restait plus que 18 000 images (sans transformation) sur les 30 000 auparavant. Il serait donc judicieux d'augmenter encore ce nombre.

3.4 Déterminations séquentielle et parallèle du genre et de l'âge

Nous avons jusqu'ici effectué les apprentissages automatiques de l'âge et du genre de manière séparée. Maintenant, il peut être intéressant d'utiliser le même réseau de neurones pour apprendre à la fois les caractéristiques de genre et d'âge. Cette idée nous provient d'un travail de Li, Liu et Chan [4] de l'Université de Hong Kong qui expliquaient que le couplage de deux tâches peut être bénéfique pour les deux tâches simultanément du fait que les informations de l'une apportaient de l'information à l'autre.

Intuitivement, les traits objectifs de vieillesse ne sont pas les mêmes que l'on soit un homme ou une femme : la forme d'une ride, sa localisation, ou même l'âge moyen de son apparition sont des facteurs qui varient en fonction du sexe de l'individu et qui sont donc des indices sur la tâche régression de l'âge. Connaître le genre de l'individu peut nous aider à interpréter la forme du visage et donc son âge. Au contraire, connaître l'âge du sujet n'est pas forcément une très bonne information pour déterminer son sexe. Si certains traits de vieillesse sont typiquement masculins ou féminins, il nous paraît évident que la tâche de classification apportera plus d'informations à la tâche de régression.

Une première idée réalisable consiste donc à effectuer séquentiellement l'apprentissage du genre et de l'âge, c'est-à-dire que l'on attend d'abord le résultat de la classification avant d'effectuer la tâche de la régression. L'implémentation de cette solution nous montre qu'elle améliore la phase d'apprentissage pour l'estimation de l'âge.

Nous avons réutilisé notre réseau convolutionnel pour l'estimation de l'âge et nous avons séparé en deux la partie avec les couches denses. Un réseau sera composé de couches denses avec

deux sorties pour la classification selon le genre et un second réseau dense se terminera par une seule sortie pour la régression de l'âge. Pour la classification, nous utilisons la fonction de perte `categorical_crossentropy` et pour la régression l'erreur quadratique moyenne (*Mean Square Error*). Les autres hyperparamètres et initialisations sont les mêmes que dans les parties précédentes.

Dans une premier temps, nous avons fait l'entraînement pour la régression de l'âge uniquement (20 itérations). Puis nous avons fait d'abord l'entraînement sur le genre (10 itérations) et ensuite celui sur l'âge (20 itérations). Dans ce cas, la partie convolutionnelle de notre réseau étant unique, les poids du réseau auront été initialisés par l'apprentissage du genre.

Voici les résultats obtenus :

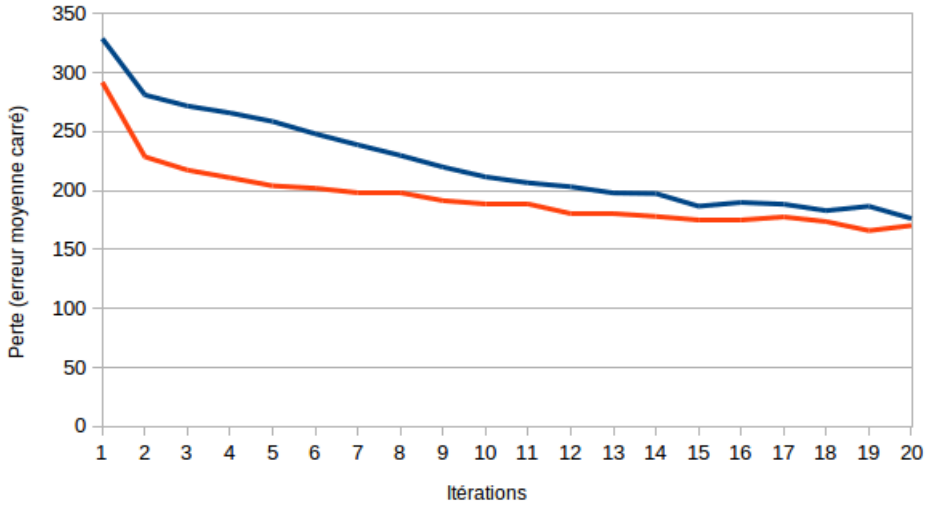

FIGURE 18 – Fonction de perte (*Mean Square Error*) : apprentissage simple pour estimer l'âge (**bleu**) et apprentissage avec un pré-entraînement sur le genre pour estimer l'âge (**rouge**)

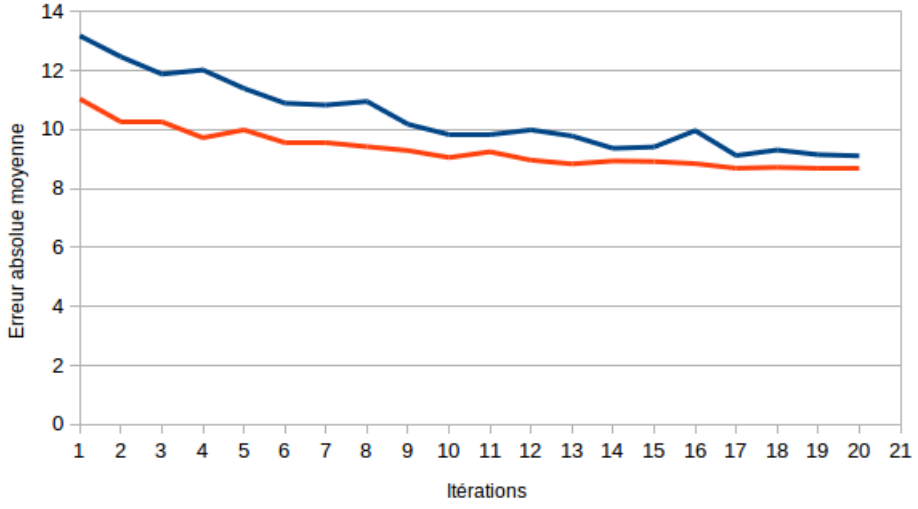

FIGURE 19 – Précision (*Mean Absolute Error*) : apprentissage simple pour estimer l'âge (**bleu**) et apprentissage avec un pré-entraînement sur le genre pour estimer l'âge (**rouge**)

Avec ce graphique, il est clairement visible que le fait de pré-entraîner le réseau pour la classification du genre aide pour l'estimation de l'âge. L'erreur initiale est plus faible au départ et reste la

plus faible durant toutes les itérations. Cependant, les deux erreurs sur l'ensemble de test pour les deux cas sont très proches. Nous pensons qu'en effectuant un apprentissage plus long ou un pré-entraînement sur le genre plus long les résultats pourraient être encore meilleurs.

L'autre solution consiste à nouveau en un réseau de neurones convolutionnel commun avec 2 MLP distincts ensuite, mais la différence ici est de faire l'entraînement de manière parallèle. Voici le schéma utilisé dans l'article de recherche.

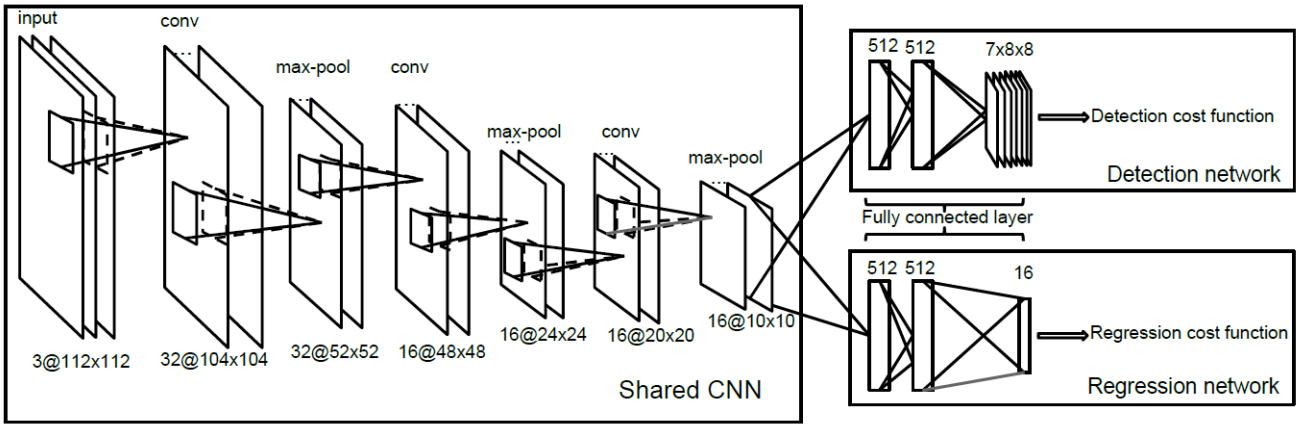

FIGURE 20 – Architecture du réseau dans l'article [4]

Malheureusement, les premiers résultats de nos simulations n'ont pas donné grand chose de concluant, et nous n'avons pas de résultat à présenter qui témoignerait d'une amélioration. Dans l'article de recherche [3], peu de détails sur l'implémentation sont donnés. Nous avons essayé différentes implémentations, principalement avec une fonction de perte jointe entre les deux MLP. Cette dernière est simplement une combinaison linéaire de la fonction de perte de la classification et de celle de la régression. Cependant aucun de nos tests n'a donné de résultats cohérents et satisfaisants. Nous pensons que c'est un problème venant de notre implémentation, et c'est le travail sur lequel nous nous pencherions si nous avions eu plus de temps.

4 Analyse critique

Nous avons décidé de travailler sur un projet concernant les réseaux profonds et principalement appliqués aux images. Le fait d'utiliser des méthodes d'apprentissage avec des images permet d'obtenir des résultats intéressants et ayant du sens. Il est facile faire le lien entre la détection ou la classification obtenues avec notre méthode avec les résultats d'un humain. Le fait d'utiliser des images en entrée de notre réseau justifie l'utilisation d'un réseau convolutionnel.

Travailler avec des images est intéressant mais soulève deux problèmes auxquels nous avons été confrontés. Le premier concerne les ressources nécessaires. En effet, les images prennent beaucoup d'espace en mémoire. De plus, même si les couches ne sont pas complètement connectées, ce qui réduit fortement le nombre de poids utilisés, nous avons été limités relativement rapidement pour l'exécution sur GPU (nos cartes graphiques ne comportent que 2 Go de mémoire vidéo). Par ailleurs, dans la plupart des articles, les méthodes proposées se basent sur des réseaux relativement profonds avec un nombre important de couches de convolutions et plusieurs couches denses. Il a donc été difficile d'essayer de réimplémenter des méthodes proposées afin de vérifier leur fonctionnement. La seconde difficulté concernait la base d'images à utiliser ainsi que les caractéristiques sur lesquelles nous pouvions faire un apprentissage. Beaucoup de bases d'images n'étaient pas accessibles publiquement, ne présentaient pas les étiquettes souhaitées ou avaient

une taille de plusieurs dizaines de Go. Après de nombreuses recherches nous avons trouvé la base IMDB-WIKI. Cette dernière était disponible directement, avait une taille de quelques Go et proposait des étiquettes intéressants. C'est la principale raison pour notre choix d'estimation de l'âge ainsi que du genre d'une personne. Par ailleurs, ces deux caractéristiques nous offraient des perspectives relativement larges : classification, régression.

Après une revue de littérature, nous avons pu tester différentes approches pour estimer le genre ainsi que l'âge. Les résultats sont intéressants mais évidemment bien moins bons que les méthodes « *state of the art* ». Enfin, nous avons voulu appliquer une approche différente à notre problème. Elle consiste en un apprentissage séquentiel ou parallèle du genre et de l'âge. Comme expliqué précédemment, cette approche a été présentée dans un article concernant une estimation de la posture d'un humain.

Finalement, ce projet nous aura permis d'approfondir notre connaissance des réseaux convolutionnels, d'effectuer une revue de littérature dans le domaine de l'estimation de caractéristiques à partir du visage (principalement âge et genre, mais aussi les émotions : sourire, tristesse, peur, etc.), et aussi d'utiliser différentes librairies de haut niveau comme Keras et TFLearn ou encore TensorFlow directement.

5 Conclusion

Au final, ce projet avait pour but de synthétiser l'utilisation de réseaux de neurones vu en classe sur un exemple concret d'utilisation : une tâche de classification et une de régression. À l'aide d'un réseau de neurone convolutionnel, suivi d'un MLP, nous avons pu inférer avec des résultats satisfaisants le genre de la personne ainsi qu'une approximation de son âge. Mieux que ça, pour la régression, nous avons pu quantifier notre intervalle de confiance en faisant l'hypothèse que nos erreurs relatives suivaient une loi normale.

Nous avons compris l'importance d'un bon ensemble de données, car c'est ce dernier qui va influencer la qualité d'apprentissage de notre algorithme. Dans notre cas, le pre-traitement semble correct car nous avons été exigeants sur la sélection des images. Néanmoins, dans l'ensemble de données et à la suite de notre sélection, nous avons favorisé l'utilisation d'hommes et de personnes ayant entre 20 et 40 ans. Nous avons également constaté une très nette amélioration de l'erreur lorsque cette personne appartenait à cette tranche d'âge. C'est pourquoi nous avons de bonnes raisons de penser qu'un ensemble de donnés plus général et mieux distribué (répartition uniforme sur l'âge plutôt que de type χ_2, et un meilleur équilibre entre hommes et femmes) résulterait en une performance bien supérieure.

De plus, nous avons tenté de diminuer l'erreur en utilisant un réseau partagé car nous sommes convaincus que l'information des deux tâches peuvent être dans une certaine mesure mutuellement profitables. Nous avons d'abord pensé que seule la classification pouvait apporter des informations supplémentaires utiles à la régression, et c'est pourquoi nous avons d'abord tenté une approche séquentielle, c'est-à-dire que l'on a d'abord détecté le genre puis estimé l'âge. Enfin nous avons réellement partagé le même sous-réseau puis exécuté en parallèle les deux tâches. Mais comme dit précédemment, nous n'avons pas obtenu de bons résultats à présenter dans ce cas-là. Comme travail futur, nous voudrions améliorer la conception de ce réseau partagé ainsi que notre implémentation, pour avoir des résultats intéressants.

Références

[1] Grigory Antipov, Sid-Ahmed Berrani, and Jean-Luc Dugelay. Minimalistic CNN-based ensemble model for gender prediction from face images. *Pattern recognition letters*, 70 :59–65, 2016.

[2] S. Jia, T. Lansdall-Welfare, and N. Cristianini. Gender Classification by Deep Learning on Millions of Weakly Labelled Images. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 462–467, Dec 2016. doi : 10.1109/ICDMW.2016.0072.

[3] Gil Levi and Tal Hassner. Age and gender classification using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 34–42, 2015.

[4] Sijin Li, Zhi-Qiang Liu, and Antoni B. Chan. Heterogeneous Multi-task Learning for Human Pose Estimation with Deep Convolutional Neural Network. *International Journal of Computer Vision*, 113(1) :19–36, 2015. ISSN 1573-1405. doi : 10.1007/s11263-014-0767-8. URL `http://dx.doi.org/10.1007/s11263-014-0767-8`.

[5] Refik Can Malli, Mehmet Aygun, and Hazim Kemal Ekenel. Apparent Age Estimation Using Ensemble of Deep Learning Models. *CoRR*, abs/1606.02909, 2016. URL `http://arxiv.org/abs/1606.02909`.

[6] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, abs/1409.1556, 2014. URL `http://arxiv.org/abs/1409.1556`.