**R**épublique Algérienne Démocratique et Populaire

**M**inistère de l'Enseignement Supérieur et de la Recherche Scientifique

**U**niversité A. Mira de Béjaïa

**F**aculté des Sciences Exactes

**D**épartement d'Informatique


# Mémoire de fin de cycle

**E**n vue de l'obtention du diplôme de master recherche en Informatique

**Option :** Systèmes d'informations avancés (SIA)

**THÈME**

# Fusion multimodale pour une éstimation précise de l'âge a partir d'images faciales

**Réalisé par :**

M$^{lle}$ ABDELLI Sonia

M$^{r}$ AFFANE Mohamed Nadjib

**Soutenu le : 01/07/2025** **Devant le jury composé de :**

| | | | |
|---|---|---|---|
| **Présidente** | M$^{me}$ | BOULAHROUZ Djamila | U.A/Mira Béjaia. |
| **Examinateur** | M$^{r}$ | ACHROUFENE Achour | U.A/Mira Béjaia. |
| **Examinateur** | M$^{r}$ | BOUCHEBBAH Fatah | U.A/Mira Béjaia. |
| **Examinatrice** | M$^{me}$ | SAAD Narimane | U.A/Mira Béjaia. |
| **Encadrant** | M$^{r}$ | KHAMMARI Mohammed | U.A/Mira Béjaia. |
| **Co-Encadrante** | M$^{me}$ | AZNI Cilia | U.A/Mira Béjaia. |

**Promotion : 2024/2025**

# Remerciements

Avant toute chose, nous exprimons notre **profonde gratitude à Dieu le Tout-Puissant**,
pour Sa guidance, Ses bienfaits et pour nous avoir donné la force, la volonté et l'inspiration nécessaires
pour mener à bien ce modeste travail.

Nous tenons à remercier chaleureusement les membres du jury
d'avoir accepté de juger et d'évaluer ce mémoire.
Nous leur sommes reconnaissants pour le temps accordé
et l'attention portée à notre travail.

Un grand merci à **M. Mohammed Khemmari**, **Mme. AZNI Cilia** nos encadrants,
pour leurs soutien précieux, leurs disponibilité, leurs confiance
et leurs conseils avisés tout au long de cette année.
leurs accompagnement patient a été essentiel
dans la réalisation de ce projet.

Nos remerciements s'adressent également à **tous les enseignants de la filière informatique**,
pour le savoir et les compétences qu'ils nous ont transmis
durant toute notre formation.

Enfin, nos remerciements les plus tendres vont à **nos familles**,
particulièrement à **nos parents**, pour leur amour inconditionnel,
leur soutien moral et matériel, ainsi que leurs encouragements constants.

*ABDELLI Sonia et AFFANE Nadjib*

# Dédicaces

*C'est avec une profonde gratitude et une vive émotion que je dédie ce travail.*

*À ma très chère mère, source intarissable d'amour, de tendresse et de prières.*
*Ta patience infinie, tes sacrifices et ton soutien indéfectible ont été ma plus grande force*
*et ma constante motivation.*
***Que Dieu te préserve et te comble de Ses bienfaits.***

*À mes frères et sœurs, pour votre affection, votre soutien et les liens précieux qui nous unissent.*
*À toute ma famille, pour vos encouragements qui m'ont toujours porté.*

*À mes amis, mes enseignants, et à tous ceux qui, par leur présence, leurs conseils*
*ou leurs encouragements, ont enrichi mon parcours et cru en moi.*

*À ma binôme, **Abdelli Sonia**, pour son engagement, sa collaboration*
*et les efforts partagés tout au long de cette aventure universitaire.*

*Et tout particulièrement, avec un amour et un respect éternels,*
*je dédie ce travail à la mémoire de mon père bien-aimé,*
***AFFANE Sofiane**.*
*Ton absence laisse un vide immense, mais ton souvenir lumineux,*
*tes valeurs, ta sagesse et l'exemple que tu as été*
*continuent d'éclairer chacun de mes pas et d'inspirer ma vie.*
*Ce travail est le fruit de l'héritage que tu m'as transmis,*
*un humble hommage à ta mémoire.*
***Que ton âme repose en paix et que la miséricorde de Dieu t'enveloppe.***

*AFFANE Nadjib*

# Dédicaces

*Avec tout mon amour, ma gratitude et mon respect,*
*je dédie ce travail à ceux qui ont compté à chaque étape de mon parcours.*

*À mon cher père et à ma précieuse mère,*
*pour leur amour inconditionnel, leurs sacrifices, leur patience et leurs prières.*
*Vous êtes ma source de force et d'inspiration.*
***Que Dieu vous protège et vous comble de Ses bienfaits.***

*À mes chères sœurs **Nabila** et **Yasmina**,*
*et à mes frères bien-aimés,*
*pour leur affection, leur soutien constant et leurs encouragements sincères.*

*À mes encadrants : **M. Mohammed Khammari** et **Mme Azni Cilia**,*
*pour leur accompagnement, leur disponibilité et leurs précieux conseils*
*tout au long de cette aventure.*

*À mes chères copines et cousines : **Dehia**, **Ludmilia** et **Yasmine**,*
*pour leur amitié sincère, leur présence réconfortante et leurs encouragements constants.*

*À mon binôme **Affane Nadjib**,*
*pour le travail partagé, la coopération et les efforts conjoints avec engagement.*

*Et à mon cher **Simo**,*
*pour son soutien, sa bienveillance et la lumière qu'il apporte à ma vie.*

*Abdelli Sonia*

# Table des matières

# Table des figures

# Liste des tableaux

# Liste des acronymes

| | | |
|---|---|---|
| **A** | | |
| | ASM | Active Shape Model |
| | AFAD | Asian Face Age Dataset |
| | AFLW | Annotated Facial Landmarks in the Wild |
| **B** | | |
| | BSIF | Binarized Statistical Image Features |
| | BN | Bacth Normalization |
| **C** | | |
| | CNN | Convolutional Neural Network |
| | CFP-FP | Celebrities in Frontal-Profile |
| | CS : Cumulative Score | |
| **D** | | |
| | Densenet | Densely Connected Network |
| | DFT | Discrete Fourier Transform |
| **E** | | |
| | ELM | Extreme Learning Machine |
| **G** | | |
| | GPU | Graphics Processing Unit |
| | GAP | Global Average Pooling |
| **I** | | |
| | IA | Intelligence Artificielle |
| | ICA | Independent Component Analysis |
| | IJB-C | IARPA Janus Benchmark - C |
| **K** | | |
| | KRR | Kernel Ridge Regression |
| | KNN | K-Nearest Neighbors |
| **L** | | |
| | LBP | Local Binary Pattern |
| | LPQ | Local Phase Quantization |
| | LPQ-GD | Local Phase Quantization with Gradient Directions |
| | LFW | Labeled Faces in the Wild |
| **M** | | |
| | MTCNN | Multi-Task Convolutional Neural Network |
| | MAE | Mean Absolute Error |
| **N** | | |
| | NMS | Non-Maxima Suppression |
| **O** | | |
| | O-Net | Output Network |
| **P** | | |
| | P-Net | Proposal Network |
| | PCA | Principal Component Analysis |
| **R** | | |
| | ResNet | Residual Network |
| | R-CNN | Region-based Convolutional Neural Network |
| | R-Net | Refine Network |
| | ReLu | Rectified Linear Unit |

| | | |
|---|---|---|
| | RGB | Red, Green, Blue |
| | ROI | Region of Interest |
| **S** | | |
| | SVM | Support Vector Machine |
| | SVR | Support Vector Regression |
| | STFT | Short-Time Fourier Transform |
| **V** | | |
| | VGG | Visual Geometry Group |
| **Y** | | |
| | YOLO | You Only Look Once. |

# Introduction générale

Ces dernières années, la vision par ordinateur connaît un développement rapide, grâce aux avancées marquantes de l'intelligence artificielle (IA), notamment dans le domaine de l'apprentissage profond (deep learning). Ce domaine permet désormais d'automatiser des tâches complexes telles que la reconnaissance faciale, la localisation d'objets ou encore l'estimation de l'âge à partir d'images. L'essentiel de ces technologies repose sur des réseaux de neurones convolutifs (CNN), qui sont capables de représenter des relations non linéaires à partir de données visuelles de grande taille[10].

L'estimation de l'âge à partir d'une image faciale repose souvent uniquement sur des informations visuelles, comme la texture de la peau ou les contours du visage. Cependant, les visages humains contiennent d'autres renseignements exploitables, comme les expressions faciales, la position, ou même des caractéristiques particulières telles que la présence de rides, de lunettes, ou la couleur des cheveux. Le principal enjeu de ce travail est donc d'examiner comment combiner ces différentes sources d'informations pour rendre l'estimation de l'âge plus précise et fiable [11].

Notre objectif est de concevoir et de mettre en œuvre un système d'estimation de l'âge basé sur l'analyse des images de visages, en exploitant des caractéristiques visuelles pertinentes extraites automatiquement à partir de réseaux de neurones profonds,afin d'obtenir une estimation d'âge plus robuste et précise.Pour atteindre cet objectif, ce travail s'articule autour des étapes suivantes :

Détection et alignement des visages : Cette première étape implique la détection de visages à l'aide de l'outil MTCNN (Multi-task Cascaded Convolutional Networks) [2],puis l'alignement et le redimensionnement de ces images.

Extraction des caractéristiques : Les images prétraitées sont ensuite soumises à une phase d'extraction des caractéristiques, en utilisant des réseaux de neurones convolutifs (CNN) tels que ResNet152v2 [3] et DenseNet121 [12], connus pour leur capacité à capturer des détails visuels complexes et à extraire des représentations vriches et discriminantes

Fusion multimodale : Les diverses sources d'information sont ensuite fusionnées en suivant une stratégie tardive, dans le but de nourrir un modèle consolidé.

Régression : Après la fusion des caractéristiques multimodales, une couche de régression basée sur un réseau de neurones est employée pour estimer la valeur cible.

Ce mémoire s'articule autour de quatre chapitres :

Dans le premier, nous proposons un état de l'art des méthodes d'estimation d'âge, couvrant

à la fois les approches des handcrafteds et les techniques modernes fondées sur l'apprentissage profond. Cette revue permet de situer les différentes approches dans leur contexte.

Ensuite, le deuxième présente une vue d'ensemble des techniques de détection de visages ainsi que des méthodes courantes d'extraction de caractéristiques et les méthodes d'apprentissage. Cette partie a pour objectif de situer le contexte théorique général, sans se limiter aux approches spécifiquement utilisées dans notre projet.

Le troisième est consacré à la présentation des différentes méthodes employées dans notre système.Nous détaillons plus particulièrement la technique adoptée pour la détection des visages, connue pour sa précision et sa robustesse dans des conditions variées,les méthodes d'extraction de caractéristiques basées sur les réseaux de neurones convolutifs (CNN),qui permettent d'extraire des représentations hautement discriminantes appelées deep features et enfin la méthode de régression. Ces méthodes constituent les étapes clés de notre système d'estimation de l'âge basé sur des images faciales.

Enfin le dernier est dédié aux tests et résultats,nous y décrivons les bases de données employées pour l'apprentissage et l'évaluation du système, ainsi que les résultats expérimentaux obtenus. Ces résultats mettent en évidence l'efficacité de notre démarche et facilitent l'évaluation de la performance de la technique suggérée dans le cadre de l'estimation d'âge via des images faciales.

# Chapitre 1

# Cadre théorique des méthodes d'estimation d'âge

## 1.1 Introduction

L'estimation de l'âge à partir d'images faciales repose fondamentalement sur deux étapes préliminaires cruciales : la détection précise du visage et l'extraction efficace de caractéristiques pertinentes . Cependant, avant même ces étapes, un prétraitement rigoureux des images est souvent nécessaire pour améliorer la qualité visuelle, normaliser les conditions d'éclairage, réduire le bruit ou encore aligner les visages détectés. La fiabilité et la précision de l'ensemble du processus biométrique dépendent étroitement de la qualité de ces traitements initiaux.

Dans ce chapitre, nous examinons les principales méthodes de détection de visages qui ont marqué l'évolution de ce domaine, en expliquant leurs principes de fonctionnement ainsi que leurs performances respectives. Nous présentons ensuite les différentes approches utilisées pour l'extraction de caractéristiques faciales , qu'elles soient basées sur des descripteurs manuels ou sur l'apprentissage profond. Enfin, nous passons en revue les méthodes d'apprentissage employées dans l'estimation de l'âge.

## 1.2 Méthodes de détection de visage

La détection de visage est une étape essentielle dans le traitement et l'analyse des images contenant des visages humains. Cette tâche implique la détection de un ou plusieurs visages dans une image et la localisation exacte de leur emplacement. Cette étape est fréquemment la première phase dans les systèmes de vision par ordinateur dédiés à l'analyse faciale, tels que l'estimation d'âge, la reconnaissance d'identité ou encore l'analyse des émotions.Pour accomplir cette tâche, plusieurs méthodes sont couramment utilisées, notamment :

### 1.2.1 Viola-Jones : Haar Cascades

La technique Viola-Jones[13], mise au point par Paul Viola et Michael Jones [13], représente l'une des premières stratégies performantes et solides pour la détection de visages en temps réel. C'est une méthode où la fonction cascade est entraînée à partir de multiples images positives (images de visages) et négatives (images sans visages), et qui est par la suite utilisée pour identifier des objets dans d'autres images. Malgré l'apparition récente de techniques fondées sur l'apprentissage profond, elle demeure largement répandue dans le domaine de la détéction de visage.

Les quatre piliers de l'algorithme Viola-Jones :

— Caractéristiques Haar-like : Il s'agit de motifs rectangulaires simples qui servent à saisir les variations de luminosité entre diverses zones du visage, telles que les yeux, le nez ou la bouche.
— image intégrale : Une technique performante qui permet de déterminer rapidement les valeurs des caractéristiques Haar-like dans l'image, en utilisant une représentation précalculée des agrégats de pixels.
— AdaBoost : c'est un algorithme d'apprentissage supervisé qui choisit les caractéristiques les plus pertinentes parmi des milliers, en fusionnant plusieurs classifieurs moins performants pour constituer un modèle global efficace.
— Classificateurs en cascade : une structure hiérarchique qui élimine progressivement les zones de l'image peu susceptibles de contenir un visage, ce qui accélère considérablement le processus de détection.

### 1.2.2 R-CNN : Regions with CNN

R-CNN (Regions with CNN features) est un modèle proposé par Ross Girshick et al [14] . Ce modèle est le premier à intégrer des réseaux de neurones convolutionnels (CNN) dans la tâche de détection d'objets. Il combine des propositions de régions, générées par l'algorithme Selective Search, avec l'extraction de caractéristiques visuelles de haut niveau. Environ 2000 régions sont extraites par image, redimensionnées à une taille fixe (227×227), puis transmises à un CNN préentraîné (comme AlexNet) pour obtenir un vecteur de caractéristiques. Ces vecteurs sont ensuite classifiés par des SVMs linéaires (un par classe), tandis qu'une régression linéaire affine la localisation des boîtes. Une étape de suppression non maximale (NMS) permet de supprimer les doublons en conservant uniquement les détections les plus pertinentes.

R-CNN a considérablement amélioré les performances de la détection d'objets, atteignant 54% mAP sur PASCAL VOC 2010 et 31.4% sur ILSVRC 2013, soit une amélioration notable par rapport aux méthodes antérieures. Cependant, son principal inconvénient est sa lenteur, car le CNN est appliqué séparément à chaque région, ce qui rend le modèle peu adapté à une utilisation en temps réel.

### 1.2.3 Faster R-CNN

Faster R-CNN est une méthode rapide et précise de détection d'objets introduite par Shaoqing Ren et al [15].Elle intègre un Réseau de Propositions de Régions (RPN) qui génère des régions candidates en glissant sur les cartes de caractéristiques d'un réseau convolutionnel, évitant ainsi les méthodes externes lentes comme Selective Search. À chaque position, il propose plusieurs boîtes (ancres) de tailles et formes différentes, en prédisant leur score d'objectivité et leurs coordonnées ajustées.

Le modèle partage ses couches convolutionnelles entre le RPN et le détecteur Fast R-CNN, ce qui réduit le coût computationnel. L'entraînement est réalisé en quatre étapes alternées pour assurer l'optimisation conjointe des deux modules. La perte combinée (classification + régression) est adaptée pour se concentrer sur les ancres pertinentes.

Faster R-CNN atteint des performances élevées (jusqu'à 73.2% mAP sur PASCAL VOC 2007) avec un temps de traitement très réduit (jusqu'à 5 images/seconde avec VGG-16), en faisant une solution efficace pour la détection d'objets en temps quasi réel.

### 1.2.4 YOLO : You Only Look Once

You Only Look Once(YOLO) est une technique de détection d'objets en temps réel présenté par Redmon et al [16] en 2016.Cette méthode reformule le problème comme une régression directe de l'image complète vers des boîtes englobantes et des probabilités associées aux classes.À l'opposé des approches classiques qui s'appuient sur des suggestions de zones (telles que R-CNN ou Faster R-CNN), YOLO exécute une unique traversée d'un réseau de neurones convolutifs pour anticiper tous les objets présents dans une image.

L'image est initialement mise à l'échelle à 448×448 pixels, puis elle est divisée en une grille S×S. Chaque cellule de ce tableau anticipe plusieurs boîtes englobantes ainsi que les probabilités associées aux différentes classes.

Chaque boîte est définie par ses coordonnées, sa dimension et un indice de fiabilité qui fusionne l'apparition d'un objet et la précision du positionnement.

Un modèle de réseau de neurones convolutif singulier produit toutes les prédictions en une unique étape, ce qui confère à celui-ci une rapidité remarquable.

Les scores sont par la suite perfectionnés, les détections peu sûres sont filtrées, puis une suppression non maximale (NMS) est mise en œuvre pour se débarrasser des répétitions.

En fin de compte, le système fournit les éléments identifiés accompagnés de leur localisation, de leur catégorie et d'un pointage associé.

### 1.2.5 MTCNN : Multi-task Cascaded Convolutional Networks

MTCNN a été élaboré pour identifier les visages et situer les points de repère (landmarks) faciaux dans des conditions non restrictives (espaces larges, angles divers, obstructions, expressions).Plutôt que de gérer la détection et l'alignement de manière distincte, Zhang et al [2]

suggèrent d'aborder ces deux missions simultanément au sein d'une structure multitâche et en cascade. Cette approche favorise l'amélioration des performances tout en préservant une vitesse d'exécution appropriée pour le temps réel.

MTCNN est constituée de trois réseaux convolutifs en cascade : le P-Net, le R-Net et l'O-Net. Ces réseaux traitent successivement l'image à différentes échelles, en utilisant une pyramide d'images, afin de détecter précisément les visages et leurs points caractéristiques.

Zhang et al [2] ont utilisé WIDER FACE et CelebA pour l'entraînement de MTCNN. WIDER FACE fournit des exemples pour la détection (positifs, négatifs, partiels), tandis que CelebA est utilisé pour les annotations des landmarks. L'évaluation est menée sur FDDB, WIDER FACE pour la détection, et AFLW pour l'alignement. Les résultats montrent que MTCNN dépasse les méthodes existantes en précision de détection et de localisation, tout en étant rapide : 99 FPS sur GPU et 16 FPS sur CPU, ce qui permet une utilisation en temps réel.

### 1.2.6 RetinaFace

RetinaFace, proposé par Jiankang Deng et al.[17] est une technique unique qui a pour but d'identifier les visages et de localiser précisément les points clés (landmarks) en un seul passage. À l'opposé des méthodes séquentielles comme MTCNN, RetinaFace offre une solution dense, rapide, et appropriée pour les visages dans des situations complexes(occlusion, forte rotation, faible résolution).

Cette méthode est basé sur l'architecture de RetinaNet, un détecteur d'objets à haute densité spécifiquement pour la détection des visages. Sa constitution inclut un backbone basé sur un réseau de neurones convolutif pré-entraîné, comme ResNet-50, ResNet-152 ou MobileNet, qui permet de déceler des attributs visuels à divers niveaux de profondeur.À cela s'ajoute un réseau de pyramide de caractéristiques (Feature Pyramid Network), qui favorise l'utilisation des caractéristiques à plusieurs échelles, essentielles pour repérer des visages de différentes dimensions.Enfin, le réseau produit plusieurs données à chaque niveau de la pyramide : les coordonnées des bounding boxes (x, y, w, h), un score de classification indiquant la probabilité de présence d'un visage, les landmarks faciaux (jusqu'à cinq points clés), ainsi qu'une régression d'ancres (anchor refinement) pour améliorer la précision des boîtes prédites.

Jiankang Deng et al. [17] utilisent WIDER FACE pour l'entraînement, enrichi par des annotations de landmarks sur 109 000 visages. L'évaluation est faite sur WIDER FACE, AFLW, AFLW2000-3D, LFW, AgeDB-30, CFP-FP et IJB-C. RetinaFace atteint 96.9 % AP (Easy), 91.8% (Hard), et une erreur de landmarks de 2.21%.

## 1.3 Extraction de caractéristiques

L'extraction de caractéristiques faciales constitue une étape essentielle dans les systèmes de reconnaissance et d'analyse faciale. Elle vise à transformer une image brute du visage en un ensemble de descripteurs numériques représentatifs, tels que les distances inter-oculaires,

la position du nez et de la bouche, ou encore des détails texturaux et des variations d'intensité locale. Ces caractéristiques capturent l'information pertinente et discriminante nécessaire pour réaliser diverses tâches, telles que l'identification individuelle, l'estimation de l'âge ou encore la détection des émotions.[18]

### 1.3.1 Méthodes d'extraction de caractéristiques basées sur les handcrafteds (shallow features)

#### 2.3.1.1 LBP (Local Binary Pattern)

Le LBP est une technique de traitement d'image utilisée pour extraire des caractéristiques locales basées sur la texture. Bien qu'il ait été initialement conçu comme un descripteur de texture général, il s'est avéré très efficace pour la reconnaissance faciale, car un visage peut être vu comme une combinaison de motifs micro-texturaux (comme la peau, les contours des yeux, etc.).[1]

L'algorithme LBP repose sur une analyse locale de chaque pixel dans une image. Voici les étapes principales :

- **Analyse d'un voisinage** $3 \times 3$ **pixels :** Pour chaque pixel central, on compare sa valeur de gris avec celles des 8 pixels environnants.
- **Seuil binaire :** Si un pixel voisin a une valeur supérieure ou égale à celle du pixel central, on attribue la valeur 1 ; sinon, 0.
- **Conversion binaire → décimale :** Les résultats des comparaisons sont organisés sous forme d'un nombre binaire de 8 bits, qui est ensuite converti en décimal.
- **Histogramme LBP :** Ce processus est répété sur toute l'image, et les valeurs LBP obtenues sont regroupées dans un histogramme représentant les caractéristiques texturales globales du visage.

Le LBP d'un pixel $(x,y)$ est défini par :

$$\mathrm{LBP}(x,y) = \sum_{n=0}^{7} 2^n \cdot s\left(l_n(x,y) - l_c(x,y)\right)$$

où :

- $l_c(x,y)$ : intensité du pixel central,
- $l_n(x,y)$ : intensité du $n^{\text{ème}}$ pixel voisin,
- $s(k) =$

$$s(k) = \begin{cases} 1 & \text{si } k \geq 0 \\ 0 & \text{sinon} \end{cases}$$

La figure 1.1 montre le fonctionnement de la méthode LBP.

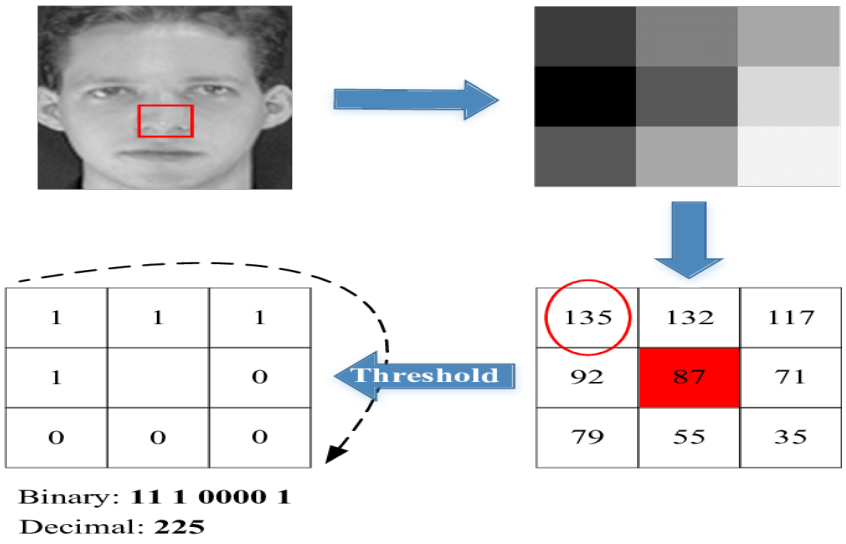


Figure 1.1 – Fonctionnement de Local binary pattern (LBP) [1]

#### 2.3.1.2 LTP (Local Ternary Patterns)

Le descripteur LTP [19] est une extension du LBP classique. Plutôt que d'utiliser une comparaison binaire simple entre le pixel central et ses voisins, LTP introduit une tolérance par le biais d'un seuil, ce qui crée trois états possibles : inférieur, proche ou supérieur. Cette ternarisation réduit la sensibilité au bruit et aux variations d'éclairage, améliorant la stabilité du descripteur dans des environnements difficiles. LTP est particulièrement efficace pour la reconnaissance faciale dans des conditions où l'éclairage est faible ou variable, ou lorsque l'image est bruitée.

#### 2.3.1.3 LPQ (Local Phase Quantization)

La méthode LPQ [20] analyse la phase locale du signal à l'intérieur d'une petite fenêtre autour de chaque pixel. Contrairement à d'autres descripteurs qui se basent sur l'amplitude ou l'intensité, LPQ tire parti des propriétés invariantes de la phase dans les images, ce qui la rend particulièrement robuste face au flou (blur) causé par des mouvements ou un mauvais focus. Concrètement, LPQ effectue une transformée de Fourier locale et quantifie les phases pour obtenir un code binaire caractéristique de la texture locale. Cela permet une reconnaissance fiable des textures même lorsque l'image est dégradée, ce qui est très utile dans des applications comme la reconnaissance faciale dans des conditions réelles.La transformée de Fourier locale discrète (DFT) est appliquée dans une fenêtre $W$ de la manière suivante :

$$F(\mathbf{u}) = \sum_{\mathbf{x} \in W} f(\mathbf{x}) \cdot e^{-j2\pi \mathbf{u}^\top \mathbf{x}}$$

où :

— $W$ est la fenêtre locale (zone d'analyse dans l'image),

— $f(\mathbf{x})$ est l'intensité du pixel à la position $\mathbf{x}$,

— **u** est le vecteur de fréquence spatiale,
— $j$ est l'unité imaginaire, telle que $j^2 = -1$.

#### 2.3.1.4 BSIF (Binarized Statistical Image Features)

BSIF[21], utilise une approche d'apprentissage statistique pour créer ses filtres, basée sur l'analyse en composantes indépendantes (ICA) appliquée à un grand ensemble de patchs naturels d'images. Cette étape permet d'apprendre un ensemble de filtres $W_i$ optimaux qui capturent les structures locales importantes.

Pour un patch d'image $X$, la réponse à chaque filtre appris $W_i$ est calculée par le produit scalaire :

$$s_i = W_i^\top X$$

Puis, chaque réponse est binarisée selon la règle suivante :

$$b_i = \begin{cases} 1, & \text{si } s_i > 0 \\ 0, & \text{sinon} \end{cases}$$

Les bits $b_i$ ainsi obtenus forment un code binaire qui caractérise la texture locale autour du patch.

En apprenant les filtres via ICA plutôt qu'en utilisant des filtres fixes, BSIF s'adapte mieux aux structures discriminantes des images, ce qui donne un descripteur puissant, compact et robuste. Cette méthode a montré de très bonnes performances dans plusieurs domaines, notamment la reconnaissance biométrique et la classification de textures.

### 1.3.2 Méthodes d'extraction de caractéristiques basées sur les CNNs (deep features)

#### 2.3.2.1 ResNet

ResNet est une architecture de réseau de neurones profonds introduite en 2015 par Kaiming He et al. [22], l'architecture de réseau de neurones profonds **ResNet** facilite l'entraînement de réseaux très profonds. Son innovation principale réside dans les blocs résiduels avec des connexions de saut (ou skip connections) qui permettent d'ajouter l'entrée du bloc à sa sortie, aidant ainsi à apprendre la fonction résiduelle.

Mathématiquement, au lieu d'apprendre directement la fonction $H(x)$, ResNet apprend la fonction résiduelle :

$$F(x) = H(x) - x$$

Ainsi, la sortie du bloc devient :

$$y = F(x) + x$$

Des variantes populaires de cette architecture existent, telles que ResNet-18 , ResNet-34 , ResNet-50 , ResNet-101 et ResNet-152 , où le nombre indique la profondeur en termes de couches entraînables. Ces modèles sont largement utilisés dans diverses tâches de vision par ordinateur.

#### 2.3.2.2 EfficientNet

EfficientNet est une architecture de réseaux convolutifs proposée par Tan et al. [23] en 2019. Elle introduit une méthode de scaling composé permettant d'ajuster simultanément la profondeur, la largeur et la résolution d'entrée du réseau de manière équilibrée. Cette approche systématique permet d'améliorer à la fois la précision et l'efficacité du modèle, contrairement aux méthodes traditionnelles qui modifient ces dimensions indépendamment les unes des autres. La base EfficientNet-B0 est conçue par recherche d'architecture neuronale (Neural Architecture Search) et utilise des blocs MBConv avec des mécanismes squeeze-and-excitation pour optimiser la représentation des caractéristiques.

#### 2.3.2.3 VGGNet

VGGNet est une architecture de réseau de neurones convolutifs proposée par Karen Simonyan et Andrew Zisserman [24] en 2014. Cette architecture se caractérise par l'utilisation répétée de petites convolutions 3×3 empilées, ce qui permet d'augmenter la profondeur du réseau tout en réduisant le nombre de paramètres par rapport à l'utilisation de filtres plus larges. Les versions les plus connues sont VGG-16 et VGG-19, qui comptent respectivement 16 et 19 couches profondes. VGGNet a obtenu d'excellents résultats lors de la compétition ImageNet 2014, démontrant que la profondeur et la simplicité des couches convolutives peuvent significativement améliorer les performances en classification d'images.

#### 2.3.2.4 Inception

Inception (ou GoogLeNet) est une architecture de réseau de neurones proposée par Szegedy et al.[25] en 2014. Elle utilise des modules Inception combinant plusieurs tailles de filtres (1×1, 3×3 ou 5×5) pour extraire des informations à différentes échelles, tout en réduisant la complexité grâce aux convolutions 1×1. Avec 22 couches, elle intègre également des classificateurs auxiliaires pour faciliter l'entraînement. Cette approche a permis d'améliorer la précision sur ImageNet tout en rendant le modèle plus léger que d'autres réseaux profonds comme VGGNet.

#### 2.3.2.5 DenseNet

DenseNet est une architecture de réseau de neurones convolutifs proposée par Gao Huang et al. [12] en 2017. Sa particularité réside dans le fait que chaque couche est connectée à toutes les

couches précédentes. Pour un réseau de $L$ couches, il existe donc un total de $\frac{L(L+1)}{2}$ connexions directes. Ainsi, chaque couche reçoit en entrée les cartes de caractéristiques de toutes les couches antérieures.

Ce mécanisme favorise la réutilisation des caractéristiques, améliore la propagation du gradient et atténue le problème de disparition du gradient. DenseNet est structuré en *blocs denses*, séparés par des *couches de transition* qui servent à réduire la dimensionnalité (via des convolutions $1 \times 1$ et des opérations de pooling).

Grâce à cette conception, DenseNet atteint de hautes performances tout en utilisant moins de paramètres et de ressources de calcul que les architectures traditionnelles. Cette efficacité a été démontrée sur plusieurs bases de données standards, notamment CIFAR-10, CIFAR-100, SVHN et ImageNet.

#### 2.3.2.6 AlexNet

AlexNet est une architecture de réseau de neurones convolutifs développée en 2012 par Alex Krizhevsky et al. [26] , cette architecture a marqué un tournant dans la reconnaissance d'images à grande échelle. Composée de cinq couches convolutionnelles suivies de trois couches entièrement connectées, elle utilise la fonction d'activation ReLU, la normalisation locale (Local Response Normalization), le « dropout » pour réduire le surapprentissage, ainsi qu'un pooling chevauchant. Grâce à l'utilisation de la puissance des GPU, AlexNet a pu être entraînée sur le vaste jeu de données ImageNet, obtenant un taux d'erreur top-5 de 15,3 %.

## 1.4 Apprentissage

En apprentissage supervisé, les algorithmes apprennent à partir de données étiquetées afin de faire des prédictions sur de nouvelles données inconnues. Parmi les tâches fondamentales de cet apprentissage, deux types de problèmes dominent : la classification et la régression . Ces deux approches diffèrent principalement par la nature de la variable cible à prédire.

### 1.4.1 La classification

La classification d'images est une tâche fondamentale en vision par ordinateur. Elle consiste à attribuer une étiquette ou une classe spécifique à une image en fonction de son contenu visuel. Ce processus repose généralement sur deux étapes principales : l'extraction de caractéristiques et la prise de décision. Lors de la première étape, des descripteurs pertinents — comme les textures, les formes ou les couleurs — sont extraits à partir des pixels bruts. Ces caractéristiques sont ensuite utilisées dans la seconde étape, où un classifieur (comme KNN, SVM ou CNN) apprend à associer ces représentations à des classes prédéfinies.

Dans le cas de la reconnaissance faciale, chaque visage est comparé à un ensemble de modèles connus dans une base de données afin d'identifier à quelle personne il correspond.[1]

### 1.4.2 La régression

La régression est une méthode statistique largement utilisée en apprentissage supervisé , dont l'objectif est de modéliser la relation entre une variable d'entrée (prédicteur) et une variable de sortie continue (cible) . Selon James et al. [27], elle permet de prédire des valeurs numériques continues à partir de données observées, comme l'âge d'une personne à partir de caractéristiques visuelles.

Dans de nombreuses applications concrètes, comme l'estimation de l'âge à partir d'images faciales ou la prédiction de la croissance économique, la régression constitue une approche essentielle pour établir des relations quantitatives à partir de données brutes.

## 1.5 Conclusion

Ce chapitre a présenté les principales étapes du processus d'estimation de l'âge à partir d'images faciales, depuis la détection du visage jusqu'aux méthodes de prédiction. Chaque composante joue un rôle essentiel dans la précision globale du système.

# Chapitre 2

# État de l'art

## 2.1 Introduction

Dans ce chapitre, nous présentons une revue des méthodes utilisées pour estimer l'âge à partir d'images faciales. Nous distinguons deux grandes familles d'approches : les méthodes basées sur les handcrafteds, qui permettent d'extraire des informations locales ou globales du visage, et les méthodes fondées sur l'apprentissage profond (deep learning), qui automatisent le processus d'extraction de caractéristiques à partir des données brutes.

## 2.2 Méthodes basées sur les handcrafteds

**Ahonen et al.** [28] proposent une méthode de reconnaissance faciale basée sur les **LBP**,un descripteur de texture robuste aux variations d'éclairage et de pose.La méthode comprend trois étapes : extraction des caractéristiques en appliquant les LBP sur des régions locales du visage, classification des descripteurs globaux via SVM ou distance chi-deux, et optimisation par sélection des régions d'intérêt. Les résultats montrent un taux de reconnaissance de 98,5% sur la base Yale et 97,5% sur ORL. Les auteurs suggèrent des améliorations futures comme l'utilisation de variantes des LBP ou l'intégration avec des CNN.

**Gunay et al.** [29] ont utilisé la méthode LBP pour classer les images de la base FERET pour estimer l'age .les classificateurs à distance minimale, plus proche voisin (nearest neighbor) et k-plus proches voisins (k-nearest neighbor) sont utilisés.Les résultat ont révélé que le système atteint une bonne précision de 80 %.

**Bekhouche et al.** [30] proposent de combiner deux descripteurs de texture : BSIF et LBP. BSIF capture des motifs statistiques globaux via des filtres appris par analyse en composantes indépendantes (ICA), tandis que LBP se concentre sur les détails locaux (bords, textures) via un codage binaire des pixels. Les caractéristiques extraites par blocs sur une région d'intérêt (ROI) du visage sont fusionnées en un vecteur d'histogrammes, puis analysées via SVR et KRR (Kernel Ridge Regression) pour prédire l'âge. Sur la base PAL, l'approche atteint une MAE (Mean Absolute Error) de 6,25 (contre 8,87 pour LBP seul) et un taux de reconnaissance de 82

% à ±5 ans. Sur FG-NET , la MAE est de 6,34 (vs 6,95 pour LBP).

**Heikkilä et Ojansivu** [31] présentent la méthode LPQ (Local Phase Quantization),Elle utilise la phase locale issue de la transformée de Fourier, qui reste stable malgré le flou symétrique. Des variantes utilisant des filtres comme STFT , Gabor , et moindres carrés , ainsi qu'une étape de décorrélation , améliorent les performances. Les tests ont été effectués sur les bases Outex TC 00001 et TC 00002. Sur images nettes, LPQgd atteint une précision de 93,9% . LBP et Gabor obtiennent chacun 90,2% . En présence de flou circulaire (rayon = 2 pixels), LPQd obtient 78,6 % de reconnaissance contre seulement 21,4 % pour LBP. Avec un flou plus fort (rayon = 5 pixels), LPQ maintient une reconnaissance à 76,5% . En cas de flou de mouvement anisotropique, la décorrélation nuit aux performances.

**Juho Kannala et Esa Rahtu** [21] proposent la méthode BSIF qui repose sur une projection linéaire de portions d'image sur un sous-espace appris à partir d'images naturelles via l'Analyse en Composantes Indépendantes (ICA), suivie d'une binarisation des coordonnées.Cette approche utilise des filtres statistiquement optimisés, améliorant la modélisation des textures naturelles. Cette méthode est testée sur les bases de données CASIA (empreintes de paume) et IITD (iris), avec des précisions allant jusqu'à 96,35% et 95,49 % respectivement.

**Xiaoyang Tan et Bill Triggs** [32] proposent une méthode améliorée pour la reconnaissance faciale en conditions d'éclairage difficiles. Elle combine un prétraitement robuste, un nouveau descripteur de texture local (LTP), une métrique de similarité basée sur la distance.En outre, la méthode utilise une extraction de caractéristiques par Kernel PCA et fusionne des indices visuels locaux issus des ondelettes de Gabor et du LBP/LTP, ce qui renforce considérablement la précision. Les résultats montrent que cette approche atteint des performances sur des bases de données difficiles comme Extended Yale-B , CAS-PEAL-R1 et FRGC-204 , avec une réduction significative du taux d'erreur. Par exemple, sur la base FRGC-204 , elle obtient un taux de vérification faciale de 88,1% avec seulement 0,1% de faux positifs , surpassant clairement les méthodes existantes.

**A.Cament et al.** [33] présentent une méthode de reconnaissance faciale robuste aux variations de pose. Elle s'appuie sur l'extraction de jets de Gabor alignés sur une grille déformable ajustée via un modèle de forme actif (ASM), et complétée par un modèle statistique local pour corriger les effets liés à la pose. La méthode intègre également une normalisation locale de l'éclairage, une pondération entropique et un vote de Borda pour la classification.Testée sur les bases FERET (±15° à ±60°) et CMU-PIE (−90° à +90°), elle atteint 100% de précision pour des poses faibles et 92,8% en moyenne sur FERET, ainsi que 87,1% sur CMU-PIE, contre 73,4% pour leur méthode antérieure. Malgré sa complexité, la méthode montre une forte robustesse aux changements de pose avec des images 2D, en faisant une solution adaptée aux applications réelles.

## 2.3 Méthodes basées sur l'apprentissage profond

**Puja Dey et al.** [34] proposent une nouvelle structure CNN pour la classification robuste de l'âge et du sexe à partir d'images en conditions réelles et non contrôlées. Le modèle CNN utilise des couches de convolution, de pooling et des couches entièrement connectées afin d'extraire et de classifier les caractéristiques. L'évaluation a été effectuée en utilisant les jeux de données Adience et UTKFace. Une répartition de 80 % pour l'apprentissage et 20 % pour le test a donné les meilleurs résultats.L'approche suggérée a surpassé les techniques précédemment établies, atteignant une précision de 86,42 % pour l'âge et de 81,96 % pour le sexe.

**He et al.** [35] introduit ResNet, une architecture de réseau de neurones profond qui transforme l'apprentissage en profondeur via l'usage de connexions résiduelles.Les résultats expérimentaux sont remarquables : sur ImageNet, ResNet-152 atteint un taux d'erreur top-5 de 3,57 %, surpassant largement GoogLeNet (6,67 %). Sur CIFAR-10, ResNet-110 obtient 96,43 % de précision, et l'architecture offre également des gains significatifs en détection d'objets sur COCO, avec une amélioration de +28 % de mAP par rapport à Fast R-CNN.

**Aruleba et al.** [36] mettent en œuvre l'architecture EfficientNet basée sur les réseaux de neurones convolutifs (CNN) pour estimer l'âge,en se servant des bases de données UTKFace et Adience. Sept variantes d'EfficientNet (B0 à B6) ont été testées dans cette étude, finement ajustées et utilisées pour évaluer l'efficacité de la classification d'âge. Les expérimentations ont montré que la variante EfficientNet-B4 offrait les meilleures performances sur les deux bases de données, avec une précision de 73,5% sur UTKFace et de 81,1 % sur Adience.

**Thaneeshan et al.** [37] suggérent une approche utilisant les réseaux de neurones convolutifs (CNN) pour l'estimation du genre et de l'âge à partir d'images de visages. Le modèle CNN a été formé et testé en s'appuyant sur la base de données Adience, qui est couramment employée dans ce secteur. Le modèle de réseau a été construit en utilisant la bibliothèque PyTorch et a subi un entraînement de 60 époques, avec un taux d'apprentissage fixé à 0,001 et une taille de lot de 32. Pour assurer la fiabilité des résultats, une méthode de validation croisée à 5 plits a été mise en œuvre. Les résultats obtenus démontrent que le modèle suggéré parvient à une exactitude de 84,20 % pour l'estimation du sexe et de 57,60 % pour l'estimation de l'âge, dépassant par conséquent plusieurs techniques existantes.Par ailleurs, une méthode de visualisation a été mise en place pour clarifier les décisions prises par le modèle, ce qui permet de déterminer les zones du visage cruciales en relation avec le genre et les tranches d'âge.

**Zaghbani et al.** [38] proposent l'utilisation d'auto-encodeurs pour faire appel aux caractéristiques apprises automatiquement par ce dernier dans un contexte supervisé pour une estimation de l'âge plus précise. L'évaluation a été réalisée en s'appuyant sur deux bases de données couramment utilisées : MORPH et FG-NET. Les données collectées indiquent que la technique est solide et efficace, affichant un taux d'erreur absolu moyen (MAE) de 3,34 % sur le MORPH database et de 3,75 % sur le FG-NET database. Ces résultats illustrent l'efficience des auto-encodeurs dans l'extraction de caractéristiques pertinentes pour la tâche d'estimation de l'âge.

**Mualla et al.** [39] proposent une technique d'évaluation de l'âge basée sur des photos de vi-

sages, en regroupant les visages selon des tranches d'âge définies. Les auteurs ont employé une approche combinant l'apprentissage profond et l'analyse en composantes principales (PCA) pour extraire et minimiser les caractéristiques propres aux images faciales.Ils ont testé l'approche sur trois jeux de données d'âges distincts provenant de Morph, et les résultats expérimentaux ont démontré une efficacité et une solidité élevées, dépassant ainsi les performances des méthodes basées sur les machines à vecteurs de support (SVM) et les k-plus proches voisins (K-NN).

**Castellano et al.** [40] combinent YOLOv5 pour la détection des visages et EfficientNet pour l'estimation de l'âge.L'ensemble du processus est automatisé : YOLOv5 identifie les visages dans l'image, ensuite EfficientNet estime l'âge à partir des zones redimensionnées. Le modèle a été formé et évalué sur le MIVIA Age Dataset, une base de données comprenant des images prises dans des conditions authentiques avec une grande diversité (lumière, expressions faciales, angles de vue). Les résultats indiquent une erreur absolue moyenne (MAE) de 2,89 ans, une performance très appréciable au regard de la complexité des données.

**Philip Smith et Cuixian Chen** [41] utilisent des réseaux de neurones profonds pré-entraînés (VGG19 et VGGFace) via le transfer learning pour la reconnaissance du genre et l'estimation de l'âge à partir d'images.Diverse méthodes d'apprentissage sont comparées, y compris la normalisation des données, l'amélioration d'images et le codage de la répartition des âges.Ils procèdent également à des tests avec une structure hiérarchique : qui commence par classifier le genre, puis met en œuvre des modèles d'âge séparés selon le genre. Les résultats affichent une précision de 98,7% pour le genre et une erreur absolue moyenne (MAE) de 4,1 ans en ce qui concerne l'âge, prouvant l'efficacité de la réutilisation de filtres CNN préexistants avec des techniques d'apprentissage modifiées.

**Shixing Chen et Caojin Zhang** [42] proposent une architecture appelée Ranking-CNN , basée sur des réseaux de neurones convolutifs (CNN). Au lieu de prédire directement l'âge comme une classe ou une valeur continue, ils utilisent $K-1$ classifieurs binaires, chacun déterminant si l'âge dépasse un seuil spécifique (par exemple >20, >30, etc.). La prédiction finale est obtenue en comptant combien de ces seuils sont franchis. Cette méthode prend en compte la nature ordinale des âges (i.e., $20 < 30 < 40$), ce qui améliore la précision. Une borne théorique de l'erreur est également fournie, ce qui guide l'entraînement du modèle et garantit sa convergence.Les expériences ont été menées sur trois bases de données bien connues : MORPH, FG-NET et Adience.La méthode proposée a obtenu une bonne performance sur ces trois bases en termes d'erreur moyenne absolue (MAE), avec des résultats respectifs de 2,92 , 4,13 et 4,4

**Niu et al.** [43] proposent une méthode d'estimation de l'âge basée sur la régression ordinale via un réseau de neurones convolutif à sorties multiples (Multiple Output CNN). Le problème est transformé en une série de tâches de classification binaire, permettant un apprentissage de bout en bout qui combine extraction de caractéristiques et prédiction.Ils utilisent deux bases de données : MORPH II (55 608 images, majoritairement africaines et européennes) et AFAD, une nouvelle base qu'ils introduisent, contenant 164 432 images de visages asiatiques âgés de 15 à 40 ans.Leur méthode obtient les meilleures performances, avec une MAE de 3.27 sur MORPH

II et 3.34 sur AFAD.

**Hu et al.** [44] mettent en place une approche innovante d'apprentissage basée sur les réseaux de neurones convolutifs profonds (CNN), en tirant parti de données faiblement annotées. Plutôt que de nécessiter des âges précis pour chaque image, ils exploitent les différences d'âge entre des paires de photos d'une même personne. Pour intégrer cette information, ils utilisent la divergence de Kullback-Leibler. De plus, ils combinent de manière adaptative deux fonctions de perte — l'entropie et l'entropie croisée — afin de guider le modèle vers une estimation d'âge précise, représentée par une distribution centrée.Par ailleurs, ils proposent une nouvelle base de données contenant plus de 100 000 visages, chacun associé à une date de prise de vue et à l'identité de la personne.

**Li et al.** [45] ont développé un modèle d'estimation de l'âge basé sur un réseau de neurones convolutif (CNN) capable d'apprendre directement à partir d'images faciales, sans intervention manuelle. Pour traiter le problème du déséquilibre des tranches d'âge dans les données, ils introduisent deux nouvelles couches : une couche cumulative, qui exploite les visages d'âges proches, et une couche de classement comparatif, qui permet d'apprendre plus efficacement les différences d'âge entre les visages. Ces améliorations rendent le modèle plus précis et permettent une utilisation optimale des données disponibles, même lorsqu'elles sont limitées.

**A.Micheal et R.Shankar** [46] proposent une méthode hybride combinant un réseau de neurones convolutif (CNN) pour l'extraction des caractéristiques faciales et un Extreme Learning Machine (ELM) pour la classification de l'âge et du genre. Cette architecture en deux étapes permet une reconnaissance à la fois plus précise et plus rapide. Testée sur la base de données Adience, composée d'images variées et non filtrées, elle atteint un taux de précision de 90% pour la classification du genre. Les résultats obtenus surpassent ceux des approches classiques, démontrant la robustesse du modèle, même en présence de données annotées de manière limitée.

## 2.4 Tableau récapitulatif des méthodes d'estimation d'âge

Nous avons résumé les méthodes citées précédemment dans le tableau 2.1.

Table 2.1 – Tableau récapitulatif de l'état de l'art

| **Auteur + Année + Réf.** | **Méthode** | **Base(s) de données** | **Précision (%)** | **MAE** |
|---|---|---|---|---|
| Ahonen et al. [28] | LBP + SVM / chi² + sélection de régions d'intérêt | Yale, ORL | 98,5 (Yale), 97,5 (ORL) | – |
| Gunay et al. [29] | LBP + Nearest Neighbor, KNN | FERET | 80 | – |

| **Auteur + Année + Réf.** | **Méthode** | **Base(s) de données** | **Précision (%)** | **MAE** |
|---|---|---|---|---|
| Bekhouche et al. [30] | BSIF + LBP + SVR/KRR | PAL, FG-NET | 82 à ±5 ans (PAL) | 6,25 (PAL), 6,34 (FG-NET) |
| Heikkilä & Ojansivu[20] | LPQ + filtres STFT, Gabor, etc. | Outex TC00001 & TC00002 | 93,9 (net), 78,6 (flou r=2), 76,5 (r=5) | – |
| Kannala & Rahtu [21] | BSIF (ICA + binarisation) | CASIA, IITD | 96,35 (CASIA), 95,49 (IITD) | – |
| Tan & Triggs [32] | LTP + KPCA + fusion LBP/Gabor | FRGC-204, Yale-B, CAS-PEAL-R1 | 88,1 (FRGC) | – |
| Cament et al. [33] | Jets de Gabor + ASM + vote Borda | FERET, CMU-PIE | 92,8 (FERET moy), 87,1 (CMU-PIE) | – |
| Puja Dey et al. [34] | CNN (âge + sexe) | Adience, UTK-Face | 86,42 (âge), 81,96 (sexe) | – |
| He et al. [22] | ResNet-152 (deep residual CNN) | ImageNet, CIFAR-10 | 96,43 (CIFAR-10) | 3,57 (ImageNet) |
| Aruleba et al. [36] | EfficientNet (B0–B6) | UTKFace, Adience | 73,5 (UTKFace), 81,1 (Adience) | – |
| Thaneeshan et al. [37] | CNN (PyTorch) + cross-val. 5-fold | Adience | 57,6 (âge), 84,2 (sexe) | – |
| Zaghbani et al. [38] | Auto-encodeurs (apprentissage supervisé) | MORPH, FG-NET | – | 3,34 (MORPH), 3,75 (FG-NET) |
| Mualla et al. [39] | Deep learning + PCA | MORPH (3 sous-ensembles) | Supérieure à SVM/KNN | – |
| Castellano et al. [40] | YOLOv5 (détection) + EfficientNet (âge) | MIVIA Age Dataset | – | 2,89 |
| Smith & Chen [41] | VGG19/VGGFace + transfer learning | – | 98,7 (sexe) | 4,1 |

| **Auteur + Année + Réf.** | **Méthode** | **Base(s) de données** | **Précision (%)** | **MAE** |
|---|---|---|---|---|
| Chen & Zhang [42] | Ranking-CNN (K–1 classifieurs binaires) | MORPH, FG-NET, Adience | – | 2,92 (MORPH), 4,13 (FG-NET), 4,4 (Adience) |
| Niu et al. [43] | Multiple Output CNN (classification ordinale) | MORPH II, AFAD | – | 3,27 (MORPH II), 3,34 (AFAD) |
| Hu et al. [44] | CNN + divergence KL + pertes entropie | Base propriétaire (>100k visages) | – | – |
| Li et al. [45] | CNN + couches cumulatives + comparatives | – | – | – |
| Micheal & Shankar [46] | CNN + Extreme Learning Machine | Adience | 90 (genre) | – |

## 2.5 Conclusion

Dans ce chapitre, nous avons présenté une synthèse des méthodes utilisées pour estimer l'âge à partir d'images faciales. Cette analyse nous permet d'identifier les forces et les faiblesses des méthodes actuelles et de définir les orientations de notre travail.

# Chapitre 3

# Approche proposée

## 3.1 Introduction

Dans ce chapitre, nous présentons les étapes clés de notre système. Une première étape consiste à détecter les visages dans une image. Le visage détecté est ensuite redimensionné et aligné afin de préparer l'image pour l'extraction des caractéristiques.Une étape d'augmentation de données est appliquée pendant l'entraînement pour améliorer la robustesse du modèle.Pour l'extraction des caractéristiques, nous utilisons deux réseaux de neurones profonds,qui fournissent des représentations riches et discriminantes. Les vecteurs extraits sont ensuite fusionnés et utilisés comme entrée à un modèle de régression permettant d'estimer l'âge de la personne.

La figure 3.1 montre le schéma général de notre système.

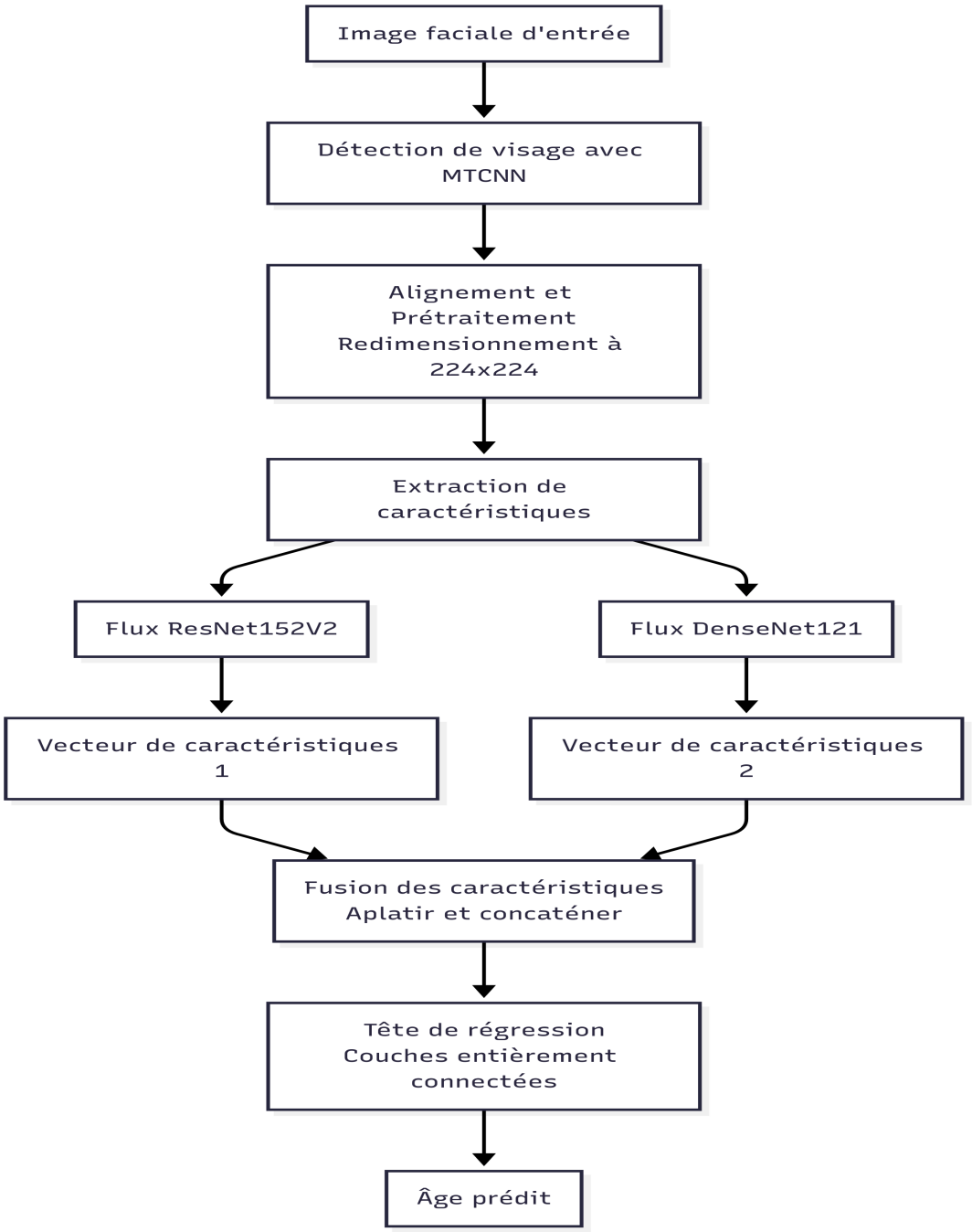


Figure 3.1 – Schéma de notre système

## 3.2 Méthode de détection de visage utilisée

Dans le cadre de notre projet,nous avons choisi le modèle MTCNN pour la détection de visage en raison de sa grande précision et sa robustesse. Ce modèle permet non seulement de détecter les visages avec une grande fiabilité, même en présence de variations de pose, d'éclairage ou d'occlusions, mais aussi d'identifier les points clés du visage (yeux, nez, bouche). Grâce à son architecture CNN en cascade, MTCNN offre des résultats stables et reproductibles, ce qui en fait une solution adaptée à notre système.

### 3.2.1 MTCNN (Multi-Task Convolutional Neural Network)

La méthode **MTCNN** [2] est une approche très efficace pour la détection et l'alignement des visages dans les images. Elle repose sur une architecture en cascade composée de trois réseaux de neurones convolutifs : **P-Net**, **R-Net** et **O-Net**, qui s'exécutent successivement afin d'améliorer progressivement la précision des résultats.

Ce qui distingue MTCNN, c'est sa capacité à combiner trois tâches simultanément :

— classifier si une région de l'image contient un visage ou non ;

— Prédire les contours du visage (bounding box) ;

— Localiser les points clés du visage comme les yeux, le nez et les coins de la bouche (landmarks)

Ces trois tâches sont liées, et en les apprenant ensemble (apprentissage multi-tâches ), la méthode arrive à de meilleures performances que si on les traitait séparément.

La figure 3.2 montre l'architecture générale de la méthode MTCNN.

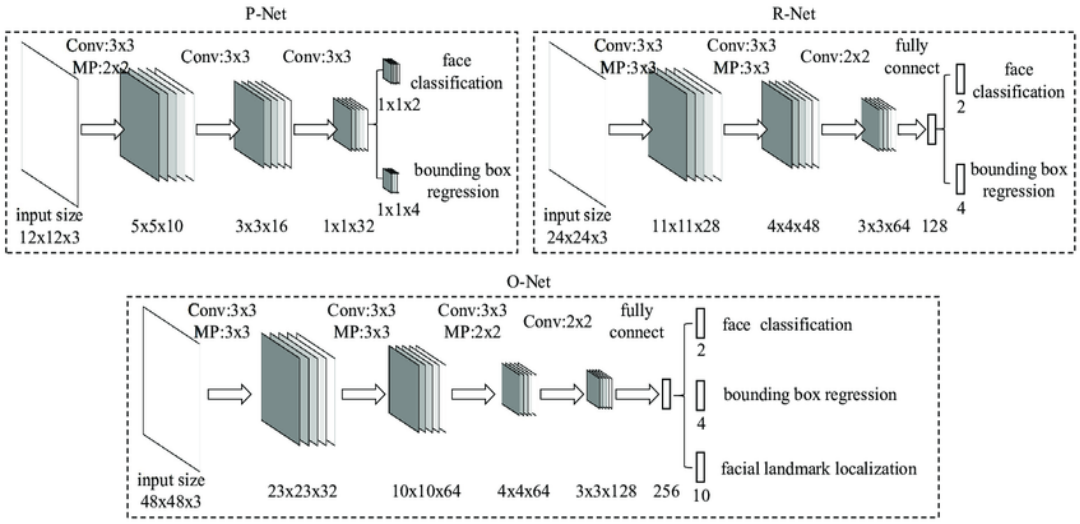


Figure 3.2 – Architecture de MTCNN [2]

#### 3.2.1.1 Bloc P-Net (Proposal Network)

Le P-Net est le premier réseau de la cascade MTCNN. Il reçoit en entrée l'image d'origine et la redimensionne à plusieurs échelles (construction d'une pyramide d'images) afin de détecter des visages de différentes tailles.

À chaque niveau d'échelle, un réseau de neurones convolutif balaie l'image à l'aide d'une fenêtre glissante (par exemple, $12 \times 12$ pixels). Pour chaque zone analysée, le P-Net effectue deux prédictions :

— la probabilité qu'une région contienne un visage (classification binaire),

— Une boîte englobante approximative autour du visage (régression des coordonnées).

Ensuite, une étape de suppression non-maximale (NMS) est appliquée pour éliminer les détections redondantes qui se chevauchent fortement.

#### 3.2.1.2 Bloc R-Net (Refine Network)

Le R-Net est le deuxième réseau de la cascade MTCNN. Son rôle est de valider et d'affiner les visages candidats proposés par le P-Net. Contrairement à ce dernier, qui est principalement optimisé pour la vitesse, le R-Net utilise une architecture plus profonde pour une analyse plus précise.

Chaque proposition issue du P-Net est redimensionnée à $24 \times 24$ pixels, puis passée à travers un réseau convolutif plus complexe.

Le R-Net effectue deux tâches principales :

— Classification binaire (visage / non-visage) afin d'éliminer les fausses détections,
— Régression des boîtes englobantes pour ajuster avec précision les coordonnées des visages détectés.

Une nouvelle étape de suppression non-maximale (NMS) est ensuite appliquée pour éliminer les détections redondantes ou très proches les unes des autres.

Grâce à cette validation et à cet affinement, le R-Net améliore significativement la qualité des détections avant de les transmettre au réseau suivant, le O-Net, pour une analyse finale.

#### 3.2.1.3 Bloc O-Net (Output Network)

Le O-Net est le dernier réseau de la cascade MTCNN, et joue un rôle crucial dans l'étape finale de détection. Contrairement au P-Net (rapide mais peu précis) et au R-Net (intermédiaire), le O-Net se concentre sur une détection fine et précise.

Il prend en entrée les régions candidates améliorées par le R-Net, redimensionnées à $48 \times 48$ pixels, et les analyse à l'aide d'un réseau convolutif plus profond et complexe.

Le O-Net effectue alors trois types de prédictions :

— Classification binaire (visage / non-visage),
— Régression très fine des boîtes englobantes,
— Détection des points clés du visage (yeux, nez, bouche).

Enfin, une étape de suppression non-maximale (NMS) est appliquée pour éliminer les détections redondantes, assurant ainsi une sortie propre et unique pour chaque visage détecté.

Le O-Net termine ainsi la cascade MTCNN avec une grande précision.

La figure 3.3 montre un exemple de détéction de visage avec les 3 réseaux de la méthode MTCNN.

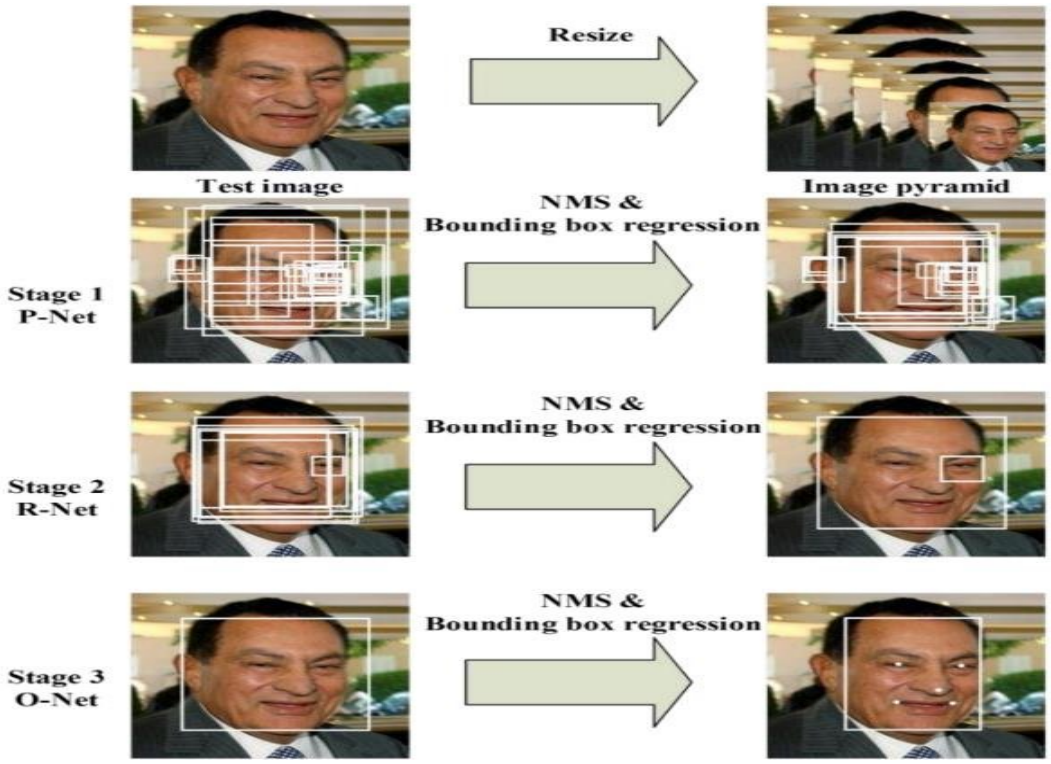


Figure 3.3 – Exemple de détection de visage avec MTCNN [2]

### 3.2.2 Alignement et redimensionnement

Après la détection des visages et la localisation des points clés (yeux, nez, bouche) par MTCNN, l'alignement est appliqué pour corriger les variations de pose. Il repose généralement sur une transformation affine basée sur la position des yeux, afin d'obtenir une orientation standard du visage. Ce dernier est ensuite redimensionné à une taille fixe de 224×224 pixels, assurant ainsi une entrée normalisée compatible avec les modèles d'extraction de caractéristiques basés sur des réseaux de neurones profonds.

## 3.3 Méthodes d'extraction de caractéristiques utilisées

Nous avons choisi les méthodes ResNet152V2 [3] et DenseNet121 [12] car ils sont largement utilisés, scientifiquement validés et reconnus pour leur performance sur des benchmarks comme ImageNet . Ces architectures sont efficaces pour capturer des détails fins dans les images. Leur utilisation repose sur des solutions éprouvées, robustes et bien documentées.

### 3.3.1 ResNet152 V2

ResNet152V2 [3] est un réseau de neurones convolutif profond appartenant à la famille des Residual Networks (ResNet), composé de 152 couches, ce qui en fait l'un des modèles CNN les plus profonds utilisés en vision par ordinateur. Contrairement à la version initiale (ResNetV1), ResNetV2 adopte une structure améliorée où les opérations de batch normalization (BN) et

d'activation ReLU sont appliquées avant les couches de convolution dans chaque bloc, ce qui améliore la stabilité de l'entraînement.

L'architecture commence par une couche de convolution initiale (7×7, stride=2) suivie d'un batch normalization et d'une activation ReLU, puis d'une couche de max-pooling (3×3, stride=2), formant ensemble le premier transition block, chargé de réduire la taille spatiale tout en augmentant la dimensionnalité des caractéristiques. Le cœur du réseau est organisé en quatre niveaux principaux (Conv2-x à Conv5-x), chacun débutant par un transition block qui change la dimensionnalité des caractéristiques et réduit la taille spatiale via une convolution 1×1 avec stride=2, suivi de plusieurs blocs bottleneck ; ces derniers sont constitués de trois couches de convolution (1×1, 3×3, 1×1), toujours précédées de batch normalization et d'activation ReLU, avec des connexions shortcut permettant de faire passer les gradients directement sans altération pour pallier au problème de vanishing gradient. Chaque niveau contient respectivement 3, 8, 36 et 3 blocs bottleneck. À la fin du réseau, une couche de Global Average Pooling (GAP) réduit les cartes de caractéristiques de dimensions spatiales élevées à une taille de 1×1 tout en conservant les canaux (généralement 2048), ce qui donne en sortie un vecteur de caractéristiques compact. Ce vecteur est ensuite transmis à une couche fully connected adaptée à la tâche ciblée (classification ou régression), souvent accompagnée d'une couche Dropout pour éviter le surapprentissage, et enfin à une fonction d'activation finale, comme softmax pour la classification multi-classes ou sigmoid/linéaire pour la régression.

La figure 3.4 montre l'architecture de la méthode ResNet152V2.

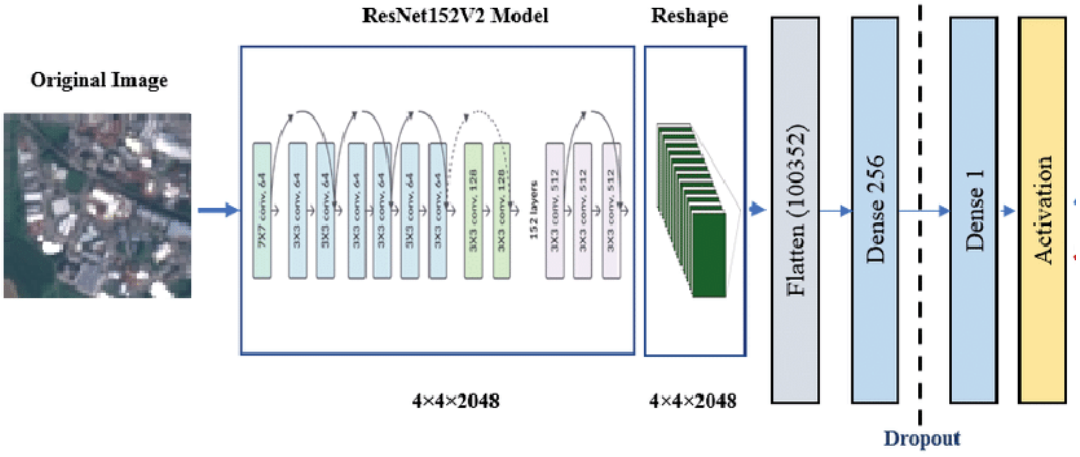


Figure 3.4 – Architecture de ResNet152V2 [3]

### 3.5.1.1 Caractéristique

Une caractéristique, c'est une information importante extraite des données comme une image, que le réseau apprend tout seul à reconnaître pendant l'entraînement. Ces caractéristiques peuvent être des détails visuels comme des bords, des textures ou des formes simples, qui aident le modèle à comprendre l'image et à effectuer des tâches. [47]

#### 3.5.1.2 Carte de caractéristique

Une carte de caractéristique est une matrice 2D qui montre comment une caractéristique spécifique est présente dans l'image d'entrée. Elle est obtenue en appliquant un filtre (ou kernel) à l'image via une opération de convolution.[47]

Chaque valeur dans la matrice indique si et à quel point la caractéristique associée au filtre est présente dans une zone particulière de l'image.

Plusieurs filtres sont utilisés en parallèle, chacun détectant une caractéristique différente (comme des bords, des textures…), ce qui donne plusieurs feature maps : chacune correspond à une caractéristique unique.

La figure 3.5 montre un exemple de carte de caractéristiques résultante d'une convolution

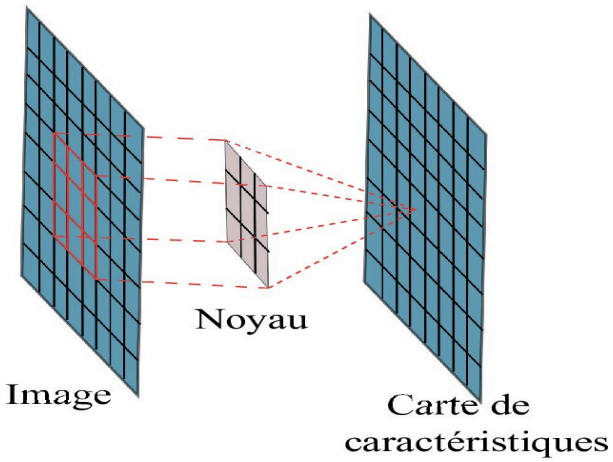


Figure 3.5 – Exemple de carte de caractéristique résutante d'une convolution [ **?** ]

#### 3.5.1.3 Connexion résiduelle (Skip Connection)

Une **s**kip connection [4], aussi appelée **connexion résiduelle**, est un concept fondamental dans les réseaux neuronaux profonds. Elle permet de faire passer directement l'entrée d'un bloc à sa sortie, en sautant une ou plusieurs couches intermédiaires.

Autrement dit, au lieu de calculer uniquement :

$$y = \mathscr{F}(x)$$

On calcule :

$$y = \mathscr{F}(x) + x$$

où :

— $x$ est l'entrée du bloc,

— $\mathscr{F}(x)$ est la transformation non linéaire (par exemple : convolutions, ReLU, BatchNorm, etc.),

— $x$ est ajouté à la sortie (c'est la **connexion de saut**).

La figure 3.6 montre un exemple de connexion résiduelle de la méthode ResNet.

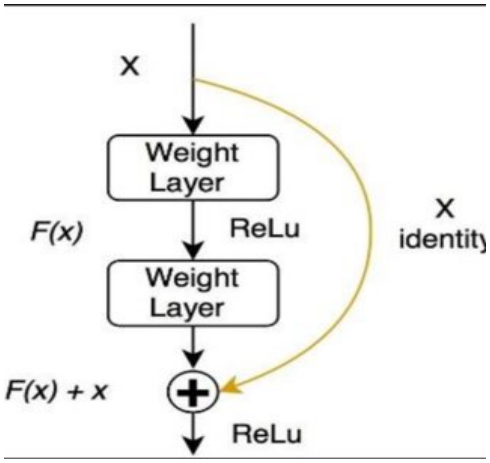


Figure 3.6 – Connexion résiduelle [4]

#### 3.5.1.4 Convolution

La convolution est une opération mathématique fondamentale utilisée dans les réseaux de neurones convolutifs (CNN) pour extraire automatiquement des caractéristiques visuelles pertinentes à partir d'images. Elle consiste à appliquer un filtre (ou noyau) sur une image ou une carte de caractéristiques, afin de détecter des motifs locaux tels que des arêtes, des textures, ou des formes plus complexes au fur et à mesure des couches profondes. [10]

##### 3.5.1.4.1 Paramètres

-Taille du filtre : Le filtre est une petite matrice qui analyse des zones locales de l'image. Les tailles courantes sont 3×3 , 5×5 ou 7×7 . Un filtre plus petit capture des détails fins avec peu de paramètres, tandis qu'un filtre plus grand couvre une zone plus large mais augmente la complexité.

-Stride : Le stride indique de combien de pixels le filtre se déplace à chaque étape. Un stride = 1 signifie un déplacement d'un pixel, alors qu'un stride = 2 réduit la taille de sortie et accélère le calcul tout en diminuant la résolution

-Padding : Le padding consiste à ajouter des valeurs autour de l'image, souvent des zéros (zero-padding ), pour garder la même taille après la convolution. Cela permet de préserver les informations aux bords et d'éviter la réduction rapide de la taille spatiale.

##### 3.5.1.4.2 Principe

Lors de l'opération de convolution , le filtre , se déplace sur l'image ou la carte de caractéristiques avec un pas (stride) défini. À chaque position, un produit scalaire est calculé entre les valeurs du filtre et celles de la région locale de l'image. Ce produit scalaire correspond à la somme des produits élément par élément entre le filtre et la portion d'image analysée. Le résultat obtenu constitue une nouvelle valeur qui sera insérée dans la carte de caractéristiques

(feature map) en sortie. En répétant cette opération sur l'ensemble de l'image, la convolution permet de détecter des motifs locaux tels que des bords, des textures ou des formes spécifiques.

La figure 3.7 montre le principe de la convolution.

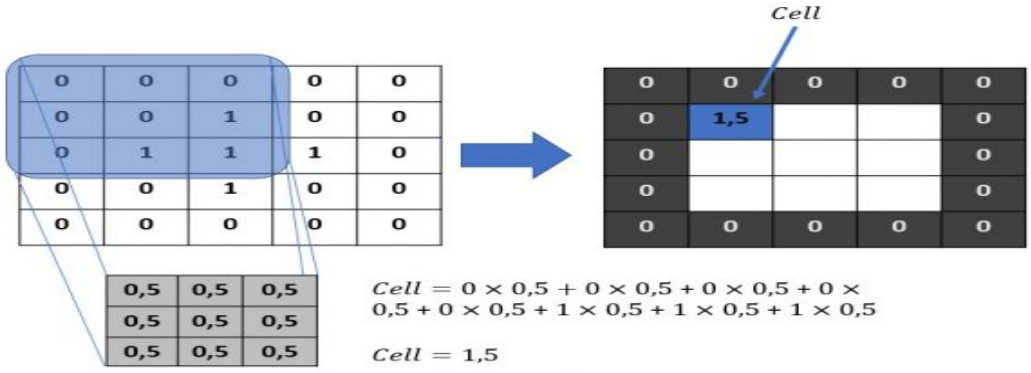


Figure 3.7 – Principe de la convolution [5]

#### 3.5.1.5 Batch normalization

La Batch Normalization (BN) [48] est une technique utilisée dans les réseaux de neurones profonds pour accélérer et stabiliser l'apprentissage . Elle permet de réduire ce qu'on appelle le Internal Covariate Shift , c'est-à-dire le changement de distribution des activations à travers les couches pendant l'entraînement, ce qui peut ralentir la convergence.

Le principe consiste à normaliser les sorties d'une couche , c'est-à-dire à centrer (moyenne = 0) et réduire (variance = 1) les valeurs des activations au sein de chaque mini-batch. Ensuite, appliquer une transformation affine avec deux paramètres appris :

$$y = \gamma \cdot \hat{x} + \beta$$

où :

— $\hat{x}$ représente l'activation normalisée (c'est-à-dire centrée et réduite),

— $\gamma$ est un paramètre d'échelle appris par le réseau,

— $\beta$ est un paramètre de décalage, également appris.

Grâce à cette étape, le réseau a la possibilité de réajuster la forme des activations si cela est nécessaire pour la tâche d'apprentissage. En résumé, cette opération permet de stabiliser l'entraînement, tout en laissant au réseau la liberté d'apprendre une représentation optimale.

#### 3.5.1.6 ReLU (Rectified Linear Unit)

La fonction ReLU [49] est l'une des fonctions d'activation les plus couramment utilisées dans les réseaux de neurones profonds. Elle est définie de manière simple par :

$$\mathrm{ReLU}(x) = \max(0, x)$$

Autrement dit :

— Si l'entrée $x > 0$, alors la sortie est égale à $x$,
— Si l'entrée $x \leq 0$, alors la sortie est égale à 0.

La figure 3.8 montre le graphe de la fonction RELU.

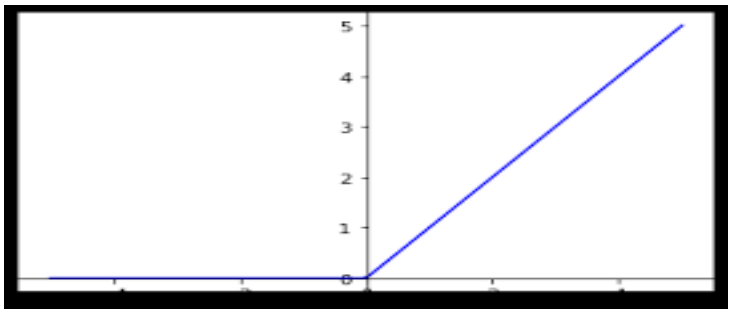


Figure 3.8 – Graphe de la fonction d'activation ReLu [6]

Cette fonction est appréciée pour sa simplicité et son efficacité : elle permet d'introduire de la non-linéarité dans le réseau tout en évitant le problème du gradient qui disparaît, courant avec d'autres fonctions comme le *sigmoïde*.

#### 3.5.1.7 Max Pooling

Le **Max Pooling** (ou sous-échantillonnage maximal) est une opération couramment utilisée dans les réseaux de neurones convolutifs (CNN). Elle permet de réduire la taille spatiale des cartes de caractéristiques tout en conservant les informations les plus importantes.

Le principe consiste à faire glisser une fenêtre (souvent de taille $2 \times 2$) sur l'image ou la carte de caractéristiques, et à **conserver uniquement la valeur maximale** dans chaque fenêtre. Cette opération permet également de réduire le surapprentissage et les besoins en calcul.

**Exemple :** Max Pooling $2 \times 2$ avec stride = 2

Entrée  Max Pooling $2 \times 2$, stride 2  Résultat

$$\begin{bmatrix} 4 & 7 & 0 & 1 \\ 6 & 9 & 2 & 5 \\ 3 & 5 & 8 & 4 \\ 2 & 1 & 3 & 6 \end{bmatrix} \quad \begin{bmatrix} \boxed{\begin{matrix} 4 & 7 \\ 6 & 9 \end{matrix}} & \boxed{\begin{matrix} 0 & 1 \\ 2 & 5 \end{matrix}} \\ \boxed{\begin{matrix} 3 & 5 \\ 2 & 1 \end{matrix}} & \boxed{\begin{matrix} 8 & 4 \\ 3 & 6 \end{matrix}} \end{bmatrix} \quad \begin{bmatrix} 9 & 5 \\ 5 & 8 \end{bmatrix}$$

#### 3.5.1.8 Global Average Pooling

Le Global Average Pooling (GAP) est une technique utilisée dans les réseaux de neurones convolutifs pour remplacer les couches fully connected traditionnellement placées à la fin du réseau. [7]

L'idée consiste à associer une carte de caractéristiques à chaque classe à prédire. Par exemple, pour une tâche de classification en 10 catégories, le modèle produit 10 cartes de caractéristiques à la sortie de la dernière couche convolutionnelle.

Au lieu de connecter ces cartes à des couches fully connected, une opération de moyennage est appliquée sur chacune. Cette opération transforme chaque carte en une seule valeur numérique. Le résultat est un vecteur dont la taille correspond au nombre de classes, et qui sert directement à la prédiction finale.

La figure 3.9 montre le vecteur résultatant du global average pooling.

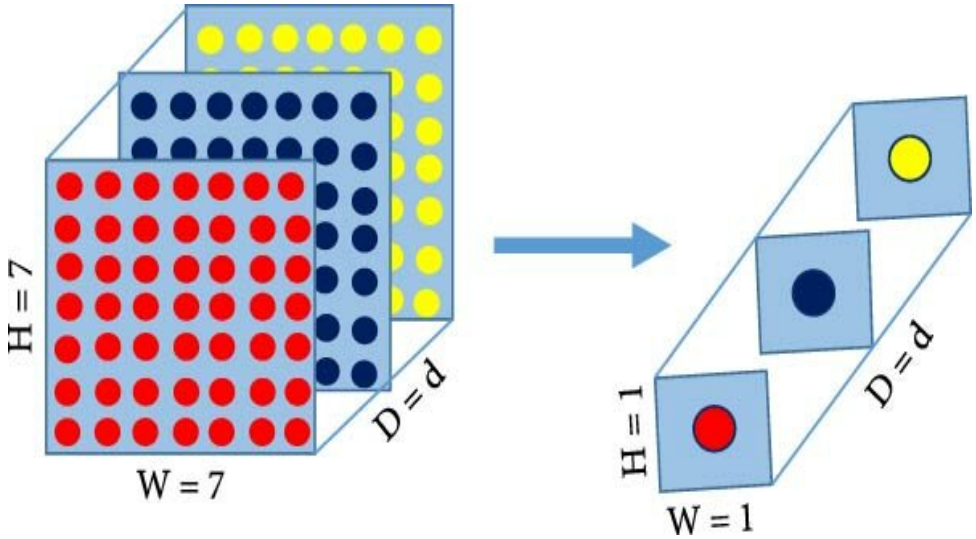


Figure 3.9 – Exemple d'un vecteur résultant du Global Average Pooling [7]

#### 3.5.1.9 Couche Flatten

Le processus d'aplatissage (ou flattening ) a pour objectif de convertir les données structurées en matrices 2D en un vecteur linéaire unique, afin de les rendre compatibles avec les couches fully connected du réseau.[8]

La figure 3.10 montre un exemple de flattening.

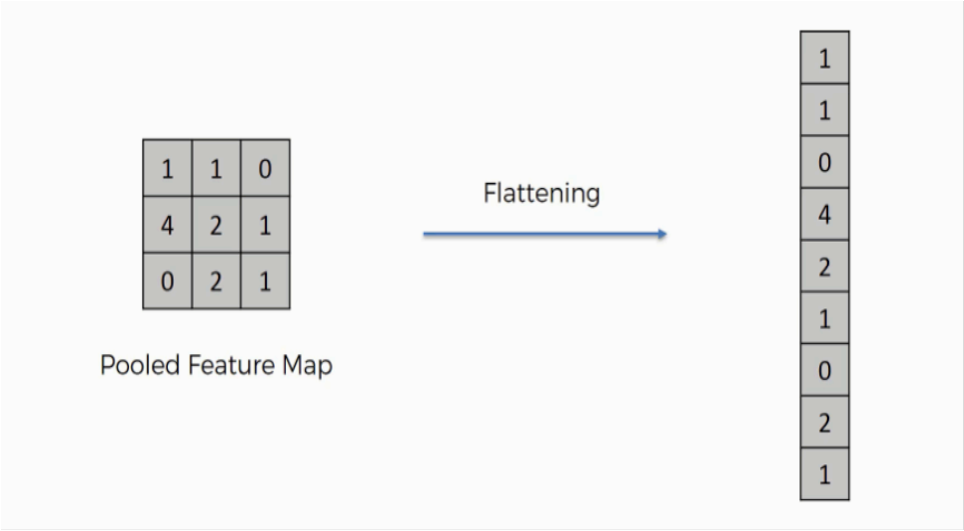


Figure 3.10 – Exemple de flattening [8]

### 3.3.2 DenseNet121

DenseNet ou Densely Connected Convolutional Networks, est une architecture innovante de réseau de neurones convolutif profond introduite par Huang et al. [12]. Elle se distingue par un concept révolutionnaire : chaque couche est connectée à toutes les autres couches suivantes. Autrement dit, la sortie d'une couche n'est pas seulement transmise à la couche immédiatement suivante (comme dans les architectures classiques), mais à toutes les couches qui viennent après elle.

Cela permet à chaque couche d'avoir accès à toutes les caractéristiques extraites précédemment, ce qui améliore considérablement le flux d'informations à travers le réseau.

L'équation clé définissant le fonctionnement de DenseNet est la suivante :

$$x_l = H_l([x_0, x_1, ..., x_{l-1}])$$

où :

- $x_l$ est la sortie de la couche $l$,

- $[x_0, x_1, ..., x_{l-1}]$ représente la concaténation des sorties de toutes les couches précédentes,

- $H_l$ est une fonction non linéaire appliquée à cette concaténation (incluant généralement une convolution, une batch normalization et une fonction d'activation comme ReLU).

Cette structure assure que chaque couche reçoit directement les informations brutes des couches antérieures, favorisant ainsi une meilleure réutilisation des caractéristiques et une propagation plus efficace du gradient pendant l'entraînement.

L'architecture de DenseNet repose sur deux éléments principaux : les Dense Blocks et les Transition Layers.

#### 3.5.2.1 Dense Block

Un Dense Block est une structure composée de plusieurs couches convolutives connectées de manière dense. Contrairement aux architectures classiques où chaque couche reçoit uniquement la sortie de la couche précédente, ici, chaque couche reçoit comme entrée l'ensemble des sorties de toutes les couches précédentes, y compris l'entrée initiale du bloc. Ces sorties sont concaténées plutôt qu'additionnées, ce qui permet de préserver l'intégralité des informations extraites à chaque étape. Par exemple, dans un bloc de quatre couches, la première couche traite uniquement l'image d'entrée, la deuxième reçoit l'image ainsi que la sortie de la première, la troisième reçoit l'image plus les sorties des deux premières couches, et ainsi de suite. Cette architecture favorise un flux d'information riche et continu, dans lequel les caractéristiques sont accumulées et enrichies progressivement, plutôt que diluées ou perdues au fil des couches.

Chaque Dense Block est constitué d'une série de blocs de convolution répétés. Chaque bloc de base comprend généralement les éléments suivants :

-Batch Normalization (BN) (voir section 3.5.1.5)

-Fonction d'activation ReLU (voir section 3.5.1.6)

-Convolution 3×3 (voir section 3.5.1.4)

#### 3.5.2.2 Transition Layer

Les Transition Layers sont placés entre les Dense Blocks pour contrôler la complexité du modèle. Ils ont trois fonctions principales :

1. Réduire la taille spatiale des caractéristiques via une opération de pooling (souvent Average Pooling),
2. Réduire le nombre de canaux features grâce à une convolution $1 \times 1$,
3. Faciliter le passage entre deux blocs denses sans perdre la cohérence des informations.

Généralement, un Transition Layer est composé successivement :

- D'une convolution $1 \times 1$ , (voir section 3.5.1.4)
- D'une couche BatchNorm, (voir section 3.5.1.5)
- D'une fonction d'activation ReLU, (voir section 3.5.1.6)
- D'un pooling $2 \times 2$ avec stride 2.

La figure 3.11 montre l'architecutre de la méthode densenet121.

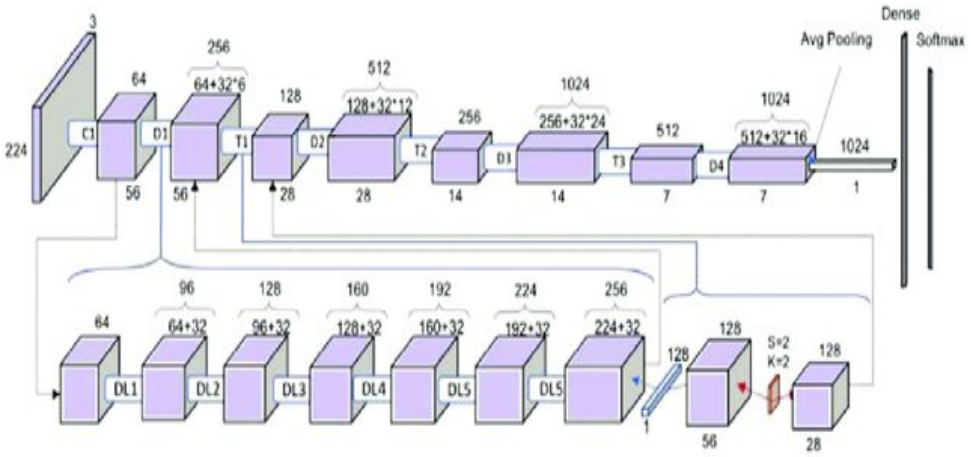


Figure 3.11 – Architecture de DenseNet121 [9]

Après l'extraction des caractéristiques à l'aide des deux architectures profondes ResNet152V2 [3] et DenseNet121[12], une opération d'aplatissage est appliquée sur les sorties globales de chaque modèle. Cela permet d'obtenir deux vecteurs de caractéristiques compacts, représentatifs de l'image d'entrée.

## 3.4 Fusion des caractéristiques extraites

Afin d'exploiter les forces complémentaires des deux modèles (ResNet152V2 et DenseNet121), nous avons choisi de fusionner les vecteurs résultant de l'opération d'aplatissage (flattening ). Pour cela, ces vecteurs sont concaténés afin de former un unique vecteur global qui intègre davantage d'informations visuelles et bénéficie de la diversité des caractéristiques extraites par les deux architectures

Dans notre système, la sortie attendue est une valeur numérique continue représentant l'âge de la personne. Pour cela, nous avons opté pour une méthode de régression, qui permet de modéliser la relation entre les caractéristiques visuelles du visage et l'âge réel de manière fluide et précise.

## 3.5 Méthode de régression utilisée

Nous avons utilisé la régression par reseau de neurones car l'âge est une variable continue. Cette méthode permet de prédire directement une valeur numérique (ex. 27,4 ans), sans classification en intervalles arbitraires. Elle est plus précise, plus naturelle et facilite l'apprentissage grâce à des métriques comme la MAE.

### 3.5.1 Régression par réseau de neurones avec une sortie linéaire

La régression par réseau de neurones est une méthode puissante utilisée pour modéliser des relations non linéaires entre des variables d'entrée et une cible continue.[50]

Après concaténation des vecteurs de caractéristiques extraits par ResNet152V2 et DenseNet121, la tête de régression se compose de :

- **Couche d'entrée** : où le nombre de neurones est égal au nombre de caractéristiques.
- **Couche cachée.**
- **Couche de sortie** : une couche dense finale à un seul neurone avec une activation linéaire permettant de produire une sortie numérique continue : l'âge prédit.

**Exemple :**

- **Entrée** : une image du visage
- **Sortie** : un nombre réel, par exemple : **32,6 ans**

## 3.6 Conclusion

Dans ce chapitre, nous avons détaillé l'ensemble des méthodes utilisées pour notre système d'estimation de l'age.A savoir le MTCNN pour la detection de visage suivie d'un alignement, d'une normalisation et d'un redimensionnement des images. Une augmentation des données a ensuite été appliquée pour améliorer la robustesse du modèle, ResNet152V2 et DenseNet121 pour l'extraction de caractéristique,enfin une régression directe.

# Chapitre 4

# Tests et résultats

## 4.1 Introduction

Ce chapitre est consacré à la présentation détaillée du processus d'entraînement, à l'analyse des performances et à la discussion des résultats obtenus par notre système. une fusion des architectures ResNet152V2 et DenseNet121, a été sélectionné pour sa capacité à extraire des caractéristiques faciales pertinentes. Nous nous attacherons ici à décrire la configuration de l'entraînement, les métriques utilisées pour l'évaluation, les performances quantitatives et qualitatives sur les bases de données de référence UTKFace et FG-NET, et enfin, une comparaison de nos résultats avec ceux de l'état de l'art et des travaux universitaires antérieurs.

## 4.2 Méthodologie d'Évaluation

### 4.2.1 Métriques d'Évaluation

Pour quantifier la performance de notre modèle d'estimation de l'âge, nous avons utilisé les métriques suivantes, classiques pour les tâches de régression :

— **Erreur Absolue Moyenne (MAE)** : Mesure la moyenne des erreurs absolues entre l'âge prédit ($\hat{y}_i$) et l'âge réel ($y_i$) sur $N$ échantillons. Une MAE plus faible indique une meilleure précision.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

— **Coefficient de Détermination ($R^2$)** : Indique la proportion de la variance des âges réels qui est expliquée par le modèle. Un score $R^2$ proche de 1 signifie un bon ajustement du modèle aux données. $\bar{y}$ est la moyenne des âges réels.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}$$

— **Score Cumulatif (CS)** : Représente le pourcentage d'images pour lesquelles l'erreur

absolue de prédiction est inférieure ou égale à un seuil $\varepsilon$ (en années). Il est calculé comme suit, où $\mathbf{1}(\cdot)$ est la fonction indicatrice :

$$\mathrm{CS}(\varepsilon) = \left( \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}(|y_i - \hat{y}_i| \leq \varepsilon) \right) \times 100\%$$

## 4.3 Préparation des Données et Apprentissage du Modèle

### 4.3.1 Prétraitement des Données

Pour l'entraînement et l'évaluation de notre système, deux bases de données principales ont été utilisées : UTKFace et FG-NET.
-La base UTKFace [51], constituée de 23705 images faciales, a été partitionnée comme suit : 15408 images pour l'ensemble d'entraînement, 4741 pour l'ensemble de validation, et 3556 pour l'ensemble de test.
-La base FG-NET[52], comprenant 1002 images, a été réservée pour évaluer la capacité de généralisation du modèle final.

Préalablement à toute phase d'apprentissage, un ensemble d'opérations de prétraitement a été appliqué uniformément à toutes les images :

1. Détection des visages au sein des images.
2. Alignement des visages détectés pour standardiser la position des caractéristiques faciales.
3. Redimensionnement de chaque visage aligné à une taille de $224 \times 224$ pixels avec 3 canaux de couleur (RGB), format requis par les architectures de réseaux de neurones convolutifs employées.
4. Augmentation de l'ensemble d'entraînement par une amplification ciblée des instances issues des classes d'âge sous-représentées, visant à équilibrer la distribution des données et à optimiser la robustesse du modèle.

Les images sont également normalisées, typiquement les valeurs des pixels sont mises à l'échelle (par exemple, entre 0 et 1), une étape courante avant de les fournir à un réseau de neurones.

### 4.3.2 Apprentissage du Modèle

**Architecture du Modèle.** Le modèle d'estimation de l'âge est basé sur une stratégie de fusion de caractéristiques issues de deux architectures CNN profondes et pré-entraînées : ResNet152V2 et DenseNet121. Pour chaque image d'entrée, les deux réseaux (sans leurs couches de classification finales) extraient des cartes de caractéristiques. Une couche de mise en commun globale moyenne est appliquée à la sortie de chaque réseau pour produire des vecteurs de caractéristiques compacts. Ces vecteurs sont ensuite concaténés, et le vecteur fusionné résultant

alimente une série de couches denses (avec 512 puis 256 neurones, activation ReLU et régularisation par Dropout à un taux de 0.4) constituant la tête de régression, qui produit finalement la prédiction de l'âge.

**Configuration de l'Entraînement.** L'entraînement du modèle a été configuré comme suit :

- **Optimiseur :** Adam, avec un taux d'apprentissage initial de $1 \times 10^{-4}$.
- **Fonction de Perte :** Erreur Absolue Moyenne (MAE), alignée sur notre principale métrique d'évaluation.
- **Gestion de l'Apprentissage :** Plusieurs mécanismes de suivi (callbacks) Keras ont été utilisés.
  - Sauvegarde des meilleurs poids : Les poids du modèle obtenant la MAE la plus faible sur l'ensemble de validation étaient systématiquement sauvegardés.
  - Arrêt anticipé : L'entraînement était stoppé si la MAE de validation ne s'améliorait pas pendant 10 époques consécutives, et les meilleurs poids sauvegardés étaient restaurés.
  - Réduction du taux d'apprentissage : Le taux d'apprentissage était réduit (facteur 0.1 ou 0.2) si la MAE de validation stagnait pendant 5 époques.
- **Générateur de Données :** Un générateur personnalisé chargeait les données par lots de 16. Pour l'ensemble d'entraînement, ce générateur appliquait un suréchantillonnage des classes d'âge sous-représentées, combiné à une augmentation de données à la volée (incluant rotations légères, zooms, retournements horizontaux) pour diversifier les exemples.
- **Durée de l'Entraînement :** Un maximum de 50 époques était prévu, l'arrêt anticipé intervenant généralement avant cette limite.
- **Environnement :** Les expérimentations ont été menées sur la plateforme Kaggle, utilisant Python, TensorFlow et Keras, avec accès à deux accélérateurs graphiques NVIDIA Tesla T4 et environ 30 Go de RAM.

## 4.4 Résultats et Analyse

Cette section présente les performances quantitatives et une analyse graphique du modèle

### 4.4.1 Performances sur la Base de Données UTKFace

L'entraînement sur UTKFace a été interrompu par le mécanisme d'arrêt anticipé à la 25$^{e}$ époque. Les poids correspondant à la meilleure MAE de validation (4.8565 ans), atteinte à l'époque 15, ont été restaurés. Les performances sur l'ensemble de test de UTKFace sont :

- **Erreur Absolue Moyenne (MAE) : 4.7076 ans**
- **Score $R^2$ : 0.8871**
- **Scores Cumulatifs (CS) :**
  - CS($\varepsilon \leq 3$ ans) : 48.14%

— CS($\varepsilon \leq 5$ ans) : 66.27%
— CS($\varepsilon \leq 10$ ans) : 87.47%

Ces résultats indiquent une bonne précision globale et un fort pouvoir explicatif du modèle sur les données UTKFace.

### 4.4.2 Analyse Graphique des Performances sur UTKFace

La Figure 4.1 montre la convergence du modèle. La MAE de validation atteint un plateau vers l'époque 15, justifiant la sauvegarde des poids à ce stade par le mécanisme de sauvegarde du modèle, avant l'intervention du mécanisme d'arrêt anticipé.

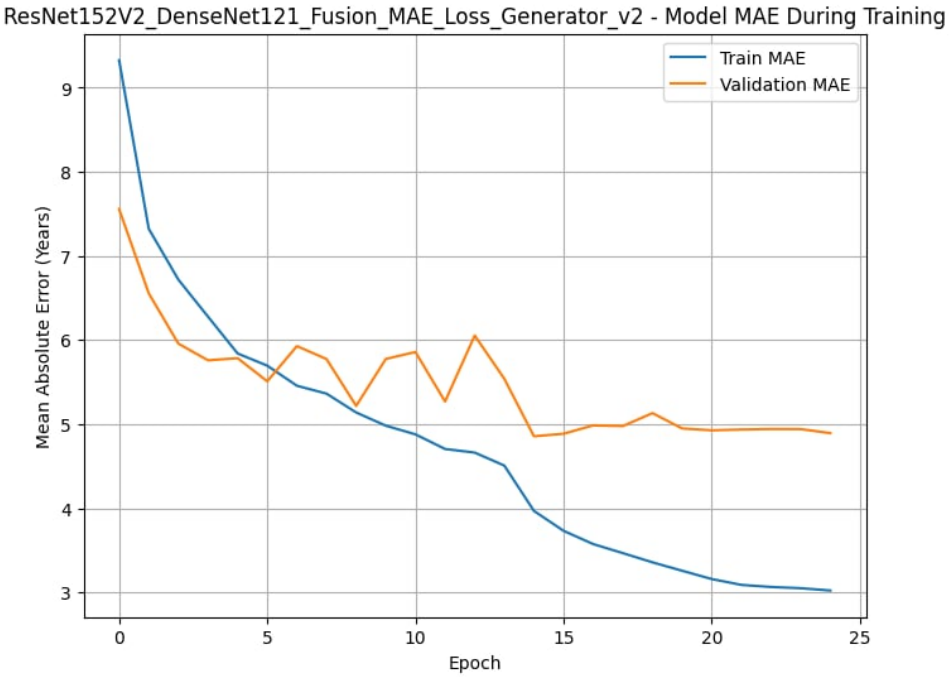


Figure 4.1 – Évolution de la MAE d'entraînement et de validation sur UTKFace. Le point optimal de validation justifie l'utilisation des mécanismes de sauvegarde et d'arrêt anticipé.

Le nuage de points (Figure 4.2) illustre une bonne corrélation entre les âges prédits et réels, bien qu'une légère sous-estimation pour les âges élevés et une dispersion pour les âges très jeunes soient observables, des défis classiques en estimation d'âge.

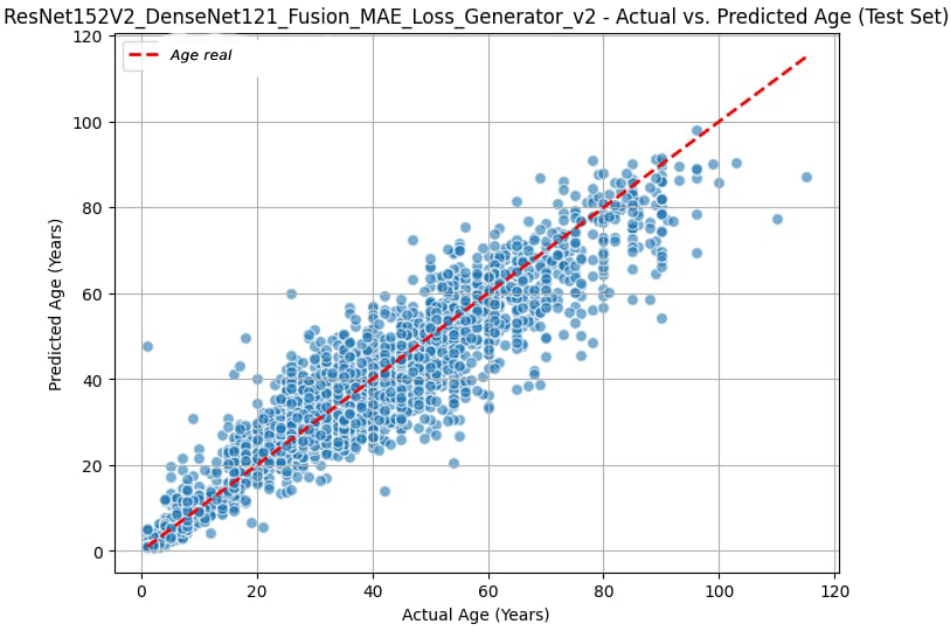


Figure 4.2 – Comparaison âge prédit vs. âge réel sur le test UTKFace.

La distribution des erreurs (Figure 4.3), centrée proche de zéro, suggère l'absence de biais systématique majeur.

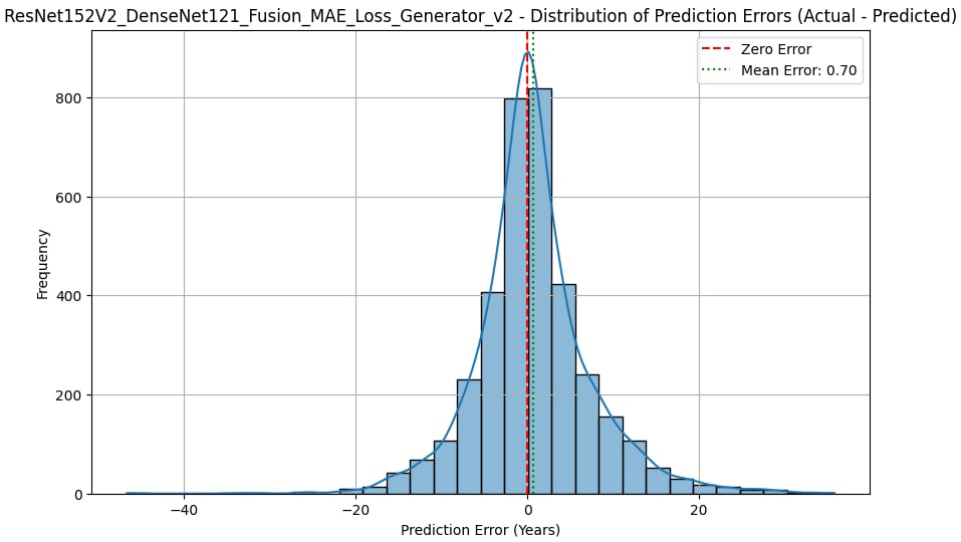


Figure 4.3 – Distribution des erreurs de prédiction (Âge Réel - Âge Prédit) sur le test UTKFace.

La courbe CS (Figure 5) confirme que la majorité des prédictions sont proches de l'âge réel, avec une montée rapide indiquant une bonne précision.

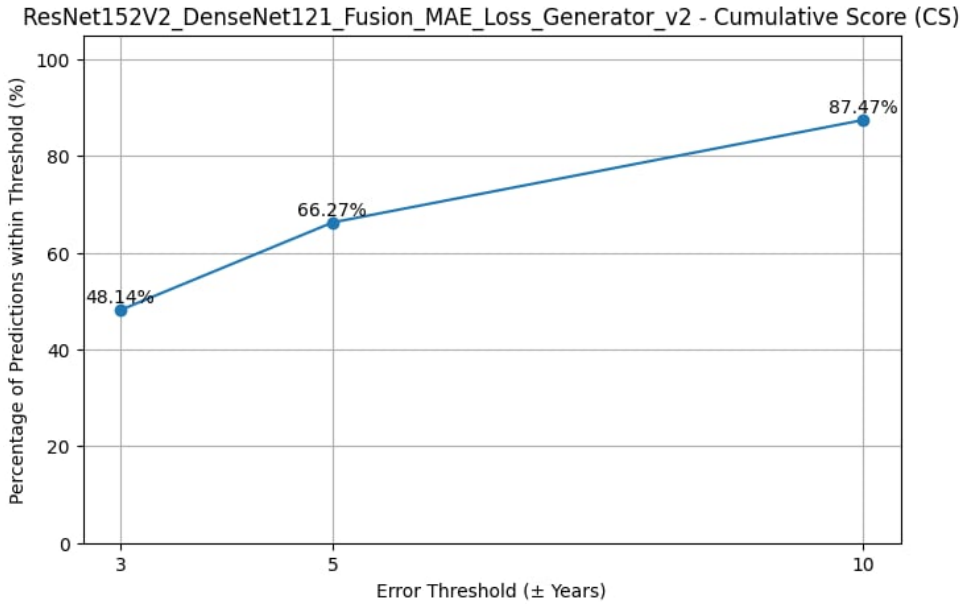


Figure 4.4 – Courbe du Score Cumulatif (CS) sur le test UTKFace.

### 4.4.3 Évaluation de la Généralisation sur la Base de Données FG-NET

Testé directement sur FG-NET (sans ré-entraînement), notre modèle entraîné sur UTKFace obtient :

- **Erreur Absolue Moyenne (MAE) : 5.1895 ans**
- **Score $R^2$ : 0.8222**
- **Scores Cumulatifs (CS) :**
  - CS($\varepsilon \leq 1$ an) : 18.06% ; CS($\varepsilon \leq 3$ ans) : 47.22% ; CS($\varepsilon \leq 5$ ans) : 63.19% ; CS($\varepsilon \leq 10$ ans) : 82.72%

La légère dégradation des performances par rapport à UTKFace est attendue en raison du décalage de domaine. Néanmoins, une MAE de 5.19 ans sur FG-NET reste compétitive et témoigne d'une bonne capacité de généralisation.

## 4.5 Comparaison et Discussion

Les performances de notre modèle ResNet152V2+DenseNet121 sont mises en perspective avec des travaux de référence et des résultats antérieurs. Les comparaisons directes doivent être interprétées avec prudence en raison des variations potentielles de protocoles.

Le tableau 4.1 illustre la comparaison des performances de notre approche avec celles d'autres méthodes.

| Méthode | Auteurs (Année) | Base(s) de données | MAE (ans) | $R^2$ | CS(5) (%) |
|---|---|---|---|---|---|
| *Méthodes de l'État de l'Art* | | | | | |
| OR-CNN [53] | Niu et al. (2016) | MORPH II, FG-NET | 3.27 (MORPH II) / 4.26 (FG-NET) | - | ∼70 (FG-NET) |
| CNN Régression (7 couches) [54] | LisanneH (2023) | UTKFace | 5.2 | - | - |
| CNN Régression Basique [55] | Dogan (2020) | UTKFace | ∼6.4 | - | - |
| CNN Baselines (sans augm.) [56] | Divers (Eur. Chem. Bull. 2023) | UTKFace | 5.0 – 6.5 | - | - |
| CRL (Ordinal Reg.) [57] | Unraveling Age Est. Puzzle (2023) | UTKFace | 5.39 | - | - |
| *Travaux Antérieurs (Université A/Mira de Bejaia)* | | | | | |
| Approche Keras [58] | AFENAI & BOUBEKRI (2022)*** | FG-NET, UTKFace | 8.6 | - | - |
| ***Notre Modèle Proposé (ResNet152V2+DenseNet121 Fusion)*** | | | | | |
| **Notre Modèle** | | **UTKFace** | **4.71** | **0.8871** | **66.27** |
| **Notre Modèle** | | **FG-NET** | **5.19** | **0.8222** | **63.19** |

Table 4.1 – Tableau comparatif des performances

**Analyse des comparaisons :**

— **Sur UTKFace :** Notre modèle (MAE 4.71 ans, $R^2$ 0.8871, CS(5) 66.27%) se positionne très favorablement. Il surpasse plusieurs approches de référence sur ce dataset, incluant LisanneH [54] (MAE 5.2), Dogan [55] (MAE ∼6.4), les baselines de [56] (MAE 5.0-6.5), et le modèle CRL [57] (MAE 5.39). Cela valide l'efficacité de la fusion de backbones profonds pré-entraînés et de la stratégie d'entraînement adoptée, incluant la gestion du déséquilibre des données.

— **Sur FG-NET :** Avec une MAE de 5.19 ans, notre modèle démontre une bonne généralisation. Il améliore substantiellement les résultats antérieurs de l'université (AFENAI & BOUBEKRI [58], MAE 8.6 ans). Bien que OR-CNN [53] obtienne une meilleure MAE (4.26 ans) sur FG-NET, cela peut être attribué à sa formulation en régression ordinale et à un éventuel pré-entraînement sur des datasets faciaux plus larges comme MORPH II.

Les résultats obtenus attestent de la robustesse et de la précision du système d'estimation d'âge proposé, basé sur la fusion des architectures ResNet152V2 et DenseNet121. La compétitivité de notre modèle, démontrée sur les datasets UTKFace et FG-NET, a justifié la valorisation de ces travaux sous la forme d'un article scientifique. Ce dernier, qui sera présenté subséquemment, expose en détail les contributions méthodologiques et les analyses comparatives approfondies menées dans le cadre de cette étude.

## 4.6 Conclusion

Dans ce chapitre, nous avons présenté en détail la conception de notre système ainsi que les différentes étapes ayant conduit à sa mise en œuvre. Nous avons décrit les algorithmes retenus, les méthodes appliquées, les outils et architectures utilisés.Les résultats obtenus sont jugés satisfaisants et montrent que le système développé est capable d'estimer l'âge des personnes avec une bonne précision. Cette approche s'est révélée efficace, confirmant la pertinence des méthodes employées.

# Conclusion générale

L'estimation d'âge à partir d'images faciales constitue un domaine riche et en constante évolution au sein de la vision par ordinateur. Dans ce contexte, notre travail s'inscrit dans le cadre de l'amélioration des performances des modèles existants grâce à une approche innovante basée sur la fusion multimodale de caractéristiques profondes extraites à partir de deux architectures convolutives performantes : ResNet152V2 et DenseNet121 . Le projet a été conçu pour répondre aux défis liés à la variabilité des visages dans des conditions non contrôlées, tels que les changements de pose, d'éclairage ou d'expression.

Dans un premier temps, nous avons mis en place un système complet de détection et de prétraitement des visages à l'aide de MTCNN (Multi-task Cascaded Convolutional Networks) . Cette étape cruciale permet de localiser précisément les visages dans les images, d'aligner les points clés (landmarks) et de redimensionner les images pour les adapter à l'entrée du modèle. Ensuite, une phase d'augmentation de données a été appliquée afin d'enrichir la base d'apprentissage et renforcer la robustesse du modèle face à différentes variations.

En ce qui concerne l'extraction de caractéristiques, nous avons choisi de combiner deux réseaux profonds reconnus pour leur efficacité dans la capture des détails pertinents dans les images : ResNet152V2 et DenseNet121. Ces modèles ont permis d'extraire des représentations riches et complémentaires des visages, qui ont ensuite été fusionnées pour former un vecteur global de caractéristiques. Ce dernier a servi à entraîner un modèle de régression linéaire simple mais efficace, capable de prédire l'âge sous forme d'une valeur continue.

Les expérimentations ont été menées sur deux bases de données largement utilisées dans le domaine : UTKFace et FG-NET . Après un processus d'entraînement rigoureux, accompagné de techniques de validation croisée et d'arrêt anticipé pour éviter le surapprentissage, le modèle final a montré des performances satisfaisantes. Sur la base UTKFace, l'erreur absolue moyenne (MAE) obtenue est de 4,71 ans , tandis que sur FG-NET, elle atteint 5,19 ans , sans fine-tuning spécifique. Ces résultats démontrent une bonne capacité de généralisation du modèle malgré les différences de distribution entre les deux ensembles de données.

Enfin, ce projet a été une opportunité non seulement de mettre en œuvre des connaissances théoriques acquises durant notre formation, mais aussi de développer des compétences pratiques

en programmation, en traitement d'image, en apprentissage profond et en analyse de données. Il représente une contribution solide dans le domaine de l'estimation d'âge faciale, tout en ouvrant la voie à des recherches futures plus ambitieuses.

Ainsi, nous espérons que cette étude servira de base à des travaux ultérieurs dans le domaine de l'intelligence artificielle appliquée à la vision par ordinateur, et qu'elle inspirera d'autres étudiants à poursuivre des projets innovants dans ce domaine en plein essor.

# Appendices

Nous avons implémenté 3 systèmes dans ce travail de recherche afin de comparer différentes approches d'extraction de caractéristiques.

-Expérience 1 : ResNet152V2 + EfficientNetB3 + LBP

Dans notre première expérience, nous avons combiné les caractéristiques extraites par deux architectures CNN performantes — ResNet150V2 et EfficientNetB3 et le handcrafted Local Binary Patterns (LBP) . Les vecteurs de caractéristiques issus des trois méthodes ont été concaténés puis utilisés comme entrée d'un modèle de régression. Cette approche a atteint une Erreur Absolue Moyenne (MAE) de 5,5 ans et un coefficient de détermination ($R^2$ ) de 0,81.

figure 5 montre le schéma récapitulatif de notre première éxpérience.

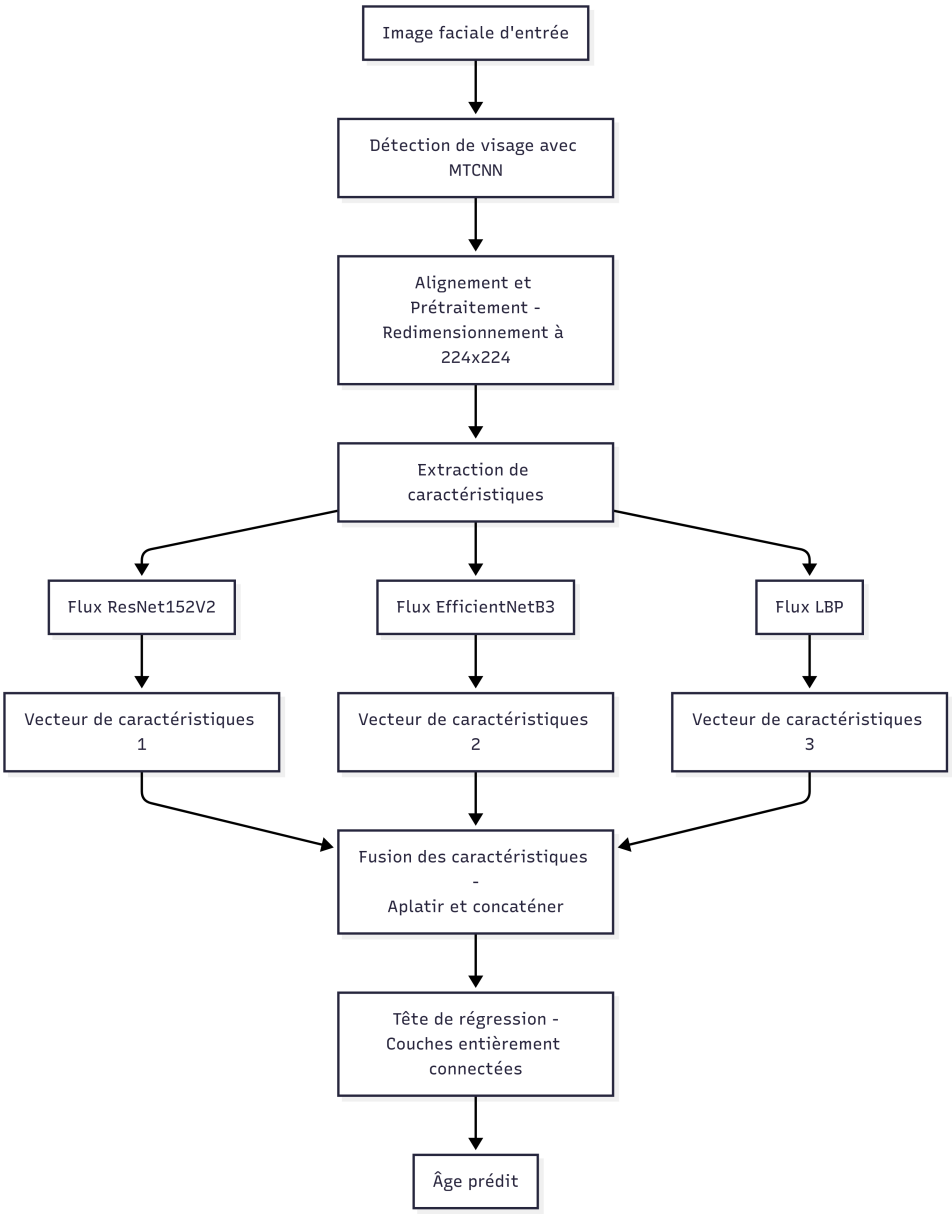


Figure 5 – Schéma du système ResNet152V2 + EfficientNetB3 + LBP

-Expérience 2 : ResNet152V2 + EfficientNetB3 (sans LBP)

Afin d'évaluer l'impact réel du descripteur LBP dans notre système, nous avons conduit une

seconde expérience en fusionnant uniquement les caractéristiques extraites par ResNet150V2 et EfficientNetB3 , sans intégrer le LBP. Les résultats obtenus ont montré une amélioration notable par rapport à la configuration précédente, avec une Erreur Absolue Moyenne (MAE) de 4,9 ans et un coefficient de détermination ($R^2$ ) de 0,84 . Ces performances suggèrent que les caractéristiques profondes apprises par les deux architectures CNN sont déjà suffisamment riches et discriminantes pour la tâche d'estimation de l'âge. Ainsi, l'intégration du LBP ne semble pas apporter d'amélioration significative dans ce cas, et pourrait même introduire une certaine redondance ou du bruit perturbateur pour le modèle. La figure 6 montre le schéma récapitulatif de notre deuxième éxpérience

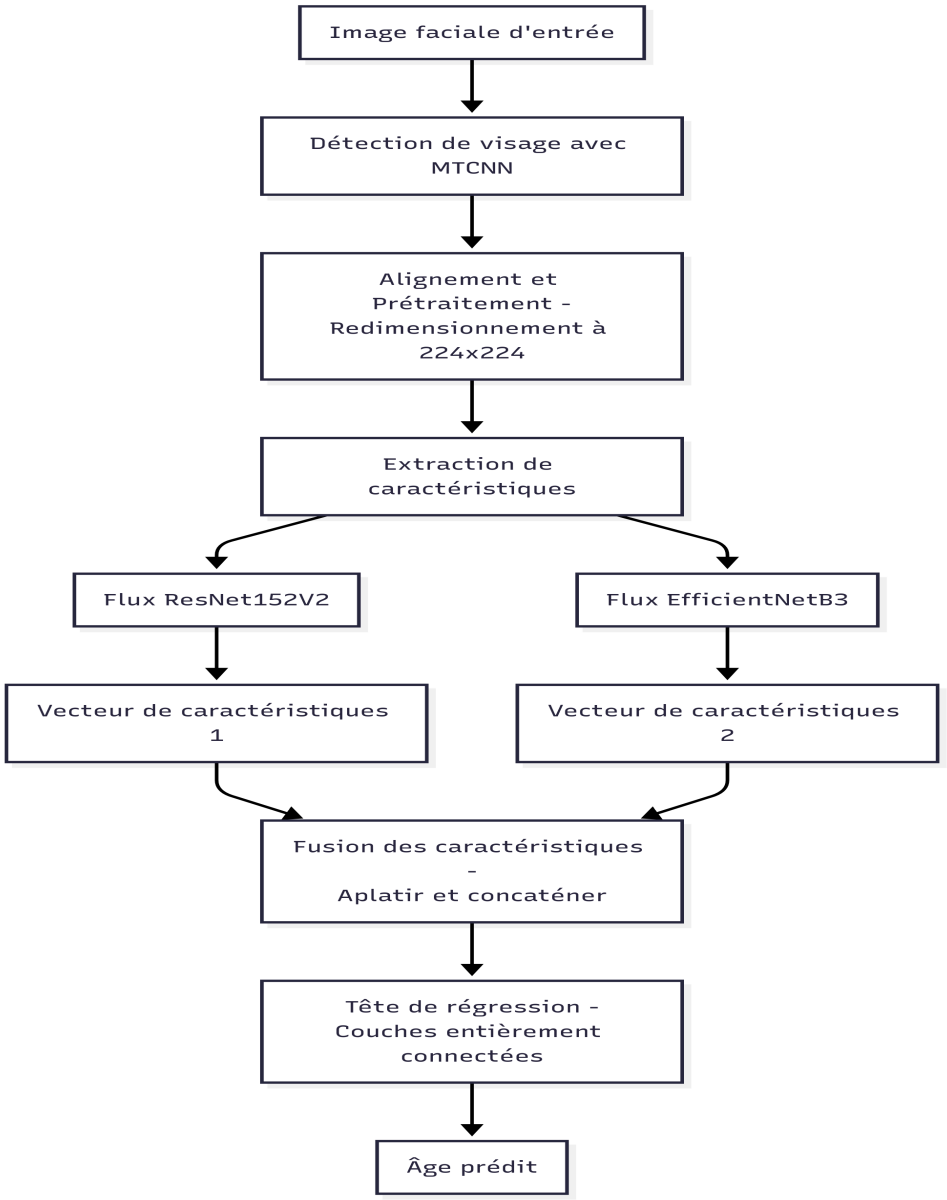


Figure 6 – Schéma du système ResNet152V2 + EfficientNetB3

La table 2 présente une comparaison de nos différentes approches expérimentales.

| **Modèle Expérimental** | **Base de données** | **MAE (ans)** | **$R^2$** | **CS(5) (%)** |
|---|---|---|---|---|
| *Expériences Préliminaires* | | | | |
| ResNet152V2 + EfficientNetB3 + LBP | UTKFace | 5.50 | 0.8100 | - |
| ResNet152V2 + EfficientNetB3 | UTKFace | 4.90 | 0.8400 | - |
| ***Modèle Final Proposé*** | | | | |
| **ResNet152V2 + DenseNet121** | **UTKFace** | **4.71** | **0.8871** | **66.27** |

Table 2 – Synthèse des performances de nos différentes architectures expérimentales.

# Annexe : Article

## Multi-modal data fusion for precise age estimation from facial images

Affane Mohamed Nadjib
*Département d'Informatique*
*Faculté des Sciences Exactes*
*Université A. Mira de Bejaia*
Bejaia, Algeria

ABDELLI Sonia
*Département d'Informatique*
*Faculté des Sciences Exactes*
*Université A. Mira de Bejaia*
Bejaia, Algeria

Mme. AZNI Cilia
*Département d'Informatique*
*Faculté des Sciences Exactes*
*Université A. Mira de Bejaia*
Bejaia, Algeria

M. KHAMMARI Mohammed
Laboratoire LIMED
*Faculté des Sciences Exactes*
*Université de Bejaia*
Bejaia, Algeria

***Abstract*—Facial age estimation is a challenging computer vision task with diverse real-world applications. Deep Convolutional Neural Networks (CNNs) have demonstrated remarkable success in this domain. This paper proposes a robust age estimation system based on the fusion of two powerful pre-trained CNN architectures: ResNet152V2 and DenseNet121. We detail our methodology, encompassing data preprocessing with MTCNN-based face detection and alignment, image normalization, and oversampling to address inherent data imbalances. The core of our system involves extracting deep features from both backbones, concatenating these features into a unified representation, and subsequently feeding them into a dedicated regression head for age prediction. Extensive experiments were conducted on the widely-used UTKFace and FG-NET datasets. Our proposed model achieves a Mean Absolute Error (MAE) of 4.71 years and an $R^2$ score of 0.8871 on the UTKFace test set, outperforming several contemporary methods. On the FG-NET dataset, the model demonstrates commendable generalization capabilities, achieving an MAE of 5.19 years and an $R^2$ score of 0.8222. These results robustly showcase the efficacy of fusing features from distinct, pre-trained deep architectures for achieving accurate facial age estimation.**

***Index Terms*—Age Estimation, Deep Learning, Convolutional Neural Networks, Feature Fusion, ResNet152V2, DenseNet121, UTKFace, FG-NET, Computer Vision.**

### I. Introduction

Facial age estimation, the process of automatically determining an individual's age from a facial image, is a prominent and actively researched subfield of computer vision and pattern recognition. Its significance stems from a wide array of practical applications, including enhanced human-computer interaction, personalized content delivery and targeted advertising, biometric security systems, and social robotics [1]. Despite its considerable potential, accurate and reliable automatic age estimation remains a formidable challenge. This difficulty arises from a confluence of factors, notably the inherent variability in the human aging process, which differs across individuals and populations. Additional complexities include diverse facial expressions, non-frontal poses, fluctuating illumination conditions, partial occlusions, and the inherently subjective nature of perceived age [2].

The advent and rapid evolution of deep learning, particularly Convolutional Neural Networks (CNNs), have revolutionized the field. CNNs have consistently demonstrated superior performance by automatically learning intricate hierarchical and discriminative features directly from raw pixel data, thereby obviating the need for laborious manual feature engineering [3], [4].

In this paper, we introduce a novel and robust facial age estimation system that synergistically leverages the complementary strengths of two state-of-the-art deep CNN architectures, ResNet152V2 and DenseNet121, through an effective feature-level fusion mechanism. The primary contributions of this work are threefold:

1) We present a systematic experimental methodology, commencing with preliminary evaluations of various CNN combinations and feature types (including LBP) to inform and justify our final model architecture selection.
2) We propose a robust fusion model that effectively combines deep features extracted from pre-trained ResNet152V2 and DenseNet121 backbones, leading to an enriched and more discriminative feature representation for age estimation.
3) We implement a comprehensive data preprocessing pipeline, featuring MTCNN-based face detection and

alignment for normalization, and an oversampling strategy to mitigate the common issue of age imbalance in training datasets.

4) We conduct extensive evaluations on the benchmark UTKFace and FG-NET datasets, demonstrating that our proposed model achieves superior or highly competitive performance compared to several existing methods and shows a significant improvement over prior related work from our institution.

The remainder of this paper is structured as follows: Section II reviews the state of the art in facial age estimation. Section III provides a detailed exposition of the proposed methodology. Section IV presents the comprehensive experimental results and analysis. Finally, Section V concludes the paper by summarizing the key findings and contributions.

## II. Related Work

Historically, age estimation methodologies often relied on meticulously hand-crafted local or global features, such as Local Binary Patterns (LBP) [5], Gabor wavelets, or Active Appearance Models (AAMs) [6]. These extracted features were then typically fed into traditional machine learning algorithms, like Support Vector Machines (SVMs) or regression models, for age prediction. However, these methods often struggled to capture the complex, non-linear aging patterns effectively.

The modern era of age estimation is dominated by deep learning. Various established CNN architectures, including VGGNet [7], ResNet [8], and DenseNet [9], have been successfully adapted and fine-tuned for the age estimation task. A particularly effective strategy involves pre-training these deep models on large-scale, general-purpose image datasets like ImageNet [10]. This pre-training allows the models to learn rich, generic visual representations, which can then be transferred and specialized for age estimation. Furthermore, advanced techniques such as ensemble methods and feature fusion—where features from multiple models or different layers within a single model are strategically combined—have shown considerable promise in further elevating predictive accuracy [11].

To contextualize our proposed model, we conducted a comparative analysis against several existing state-of-the-art methods and prior work from our institution. A summary of this comparison is provided in Table I. It is important to note that direct comparisons can sometimes be nuanced due to variations in exact preprocessing steps, dataset splits, and evaluation protocols across different studies.

The comparative analysis reveals that our proposed model demonstrates highly competitive performance. On the **UTKFace dataset**, our MAE of 4.71 years is superior to several other methods, including the 7-layer CNN by LisanneH (MAE 5.2) [12], basic CNNs (MAE 5.0–6.5) [14], and the CRL ordinal regression model (MAE 5.39) [15]. This underscores the effectiveness of our fusion architecture and preprocessing pipeline on a challenging "in-the-wild" dataset.

On the **FG-NET dataset**, our model achieves an MAE of 5.19 years. While some specialized methods like OR-CNN [4] report a lower MAE, our result is highly competitive and demonstrates strong generalization. Crucially, our model shows a substantial improvement over the prior work from our institution by AFENAI & BOUBEKRI [16] (MAE 8.6), highlighting the significant advancements of our refined methodology.

## III. Methodology

Our proposed age estimation framework is systematically structured, involving several key stages: data acquisition and meticulous preprocessing, a phase of preliminary experimentation to guide optimal model selection, the detailed design of our final feature fusion architecture, and subsequent rigorous model training and evaluation.

### A. Datasets

Two publicly accessible and widely utilized datasets were employed for training and evaluating our age estimation model:

- **UTKFace Dataset [17]:** This large-scale dataset comprises over 23,000 facial images. It offers extensive age diversity, ranging from 0 to 116 years, and includes annotations for age, gender, and ethnicity. For our primary model development and evaluation, we utilized a curated subset of 23,705 images. This subset was partitioned into 15,408 images for training, 4,741 for validation, and 3,556 for testing. UTKFace is particularly valuable as it contains images captured "in the wild," exhibiting considerable variations in pose, facial expression, and illumination.
- **FG-NET Dataset [18]:** The FG-NET dataset contains 1,002 images pertaining to 82 distinct subjects, with multiple images available per subject captured at different ages. This longitudinal characteristic, along with its varied image quality, makes FG-NET a challenging and frequently used benchmark for assessing the generalization capabilities of age estimation models. We employed FG-NET specifically to evaluate the robustness and generalization performance of our best model, which was primarily trained on the UTKFace dataset.

### B. Data Preprocessing

A rigorous and effective data preprocessing pipeline is paramount for achieving robust and accurate model performance in facial analysis tasks. The following sequential steps were meticulously applied to all images from both datasets:

1) **Face Detection:** The Multi-Task Cascaded Convolutional Network (MTCNN) [19] was employed for its proven efficacy in accurate face detection and precise localization of key facial landmarks (e.g., eyes, nose, and mouth).
2) **Face Alignment:** Leveraging the detected facial landmarks, particularly the eye centers, facial images were geometrically aligned using a similarity transformation.

TABLE I
COMPARATIVE ANALYSIS OF AGE ESTIMATION PERFORMANCE WITH STATE-OF-THE-ART METHODS AND PRIOR WORK.

| Method | Authors (Year) | Dataset(s) Evaluated | MAE (years) | $R^2$ | CS(5) (%) |
|---|---|---|---|---|---|
| *State-of-the-Art Methods* | | | | | |
| OR-CNN [4] | Niu et al. (2016) | MORPH II, FG-NET | 3.27 / **4.26** | – | ∼70 (FG-NET) |
| CNN Regression (7 layers) [12] | LisanneH (2023) | UTKFace | 5.2 | – | – |
| Basic CNN Regression [13] | Dogan (2020) | UTKFace | ∼6.4 | – | – |
| Baseline CNNs (no aug.) [14] | Various (Eur. Chem. Bull. 2023) | UTKFace | 5.0 – 6.5 | – | – |
| CRL (Ordinal Reg.) [15] | Unraveling Age Est. Puzzle (2023) | UTKFace | 5.39 | – | – |
| *Prior Work (Université A/Mira de Bejaia)* | | | | | |
| Keras Approach | AFENAI & BOUBEKRI (2022) [16] | FG-NET, UTKFace | 8.6 | – | – |
| ***Proposed Model (ResNet152V2+DenseNet121 Fusion)*** | | | | | |
| **This work** | **AFFANE et al.** | **UTKFace (test)** | **4.71** | **0.8871** | **66.27** |
| **This work** | **AFFANE et al.** | **FG-NET (test)** | **5.19** | **0.8222** | **63.19** |

Note: $R^2$ and CS(5) values are not consistently reported across all original publications. FG-NET MAE for OR-CNN is highlighted for direct comparison.

This step normalizes the facial pose and ensures a consistent spatial arrangement of facial features across images, reducing pose-induced variability.

3) **Resizing:** All aligned facial images were uniformly resized to $224 \times 224$ pixels. This dimension is a standard input requirement for many prominent pre-trained CNN architectures, including the ResNet152V2 and DenseNet121 models utilized in our work.
4) **Normalization:** Pixel intensity values, initially in the range [0, 255], were normalized to the range [0, 1] by dividing each pixel value by 255.0. This standard practice aids in stabilizing and accelerating the training process of deep neural networks.
5) **Oversampling for Data Imbalance:** Age distributions in facial datasets frequently exhibit significant imbalance, with substantially fewer samples available for very young and very elderly individuals compared to middle-aged groups. To counteract this, we implemented an oversampling technique specifically on the training set. This involved selectively replicating samples from under-represented age classes to create a more balanced age distribution for model training, thereby mitigating potential biases towards majority age groups.

All images were processed and maintained in the RGB (Red, Green, Blue) color format, providing three input channels to the CNN models.

### C. Preliminary Experiments

To inform the design choices for our final model and to establish a performance baseline, a series of preliminary experiments were conducted on the UTKFace dataset. These experiments explored different architectural combinations and feature types:

- **ResNet150V2 + EfficientNetB3 + LBP:** An initial approach involved fusing deep features extracted from ResNet150V2 and EfficientNetB3 backbones with handcrafted Local Binary Pattern (LBP) texture features. This hybrid model yielded a Mean Absolute Error (MAE) of 5.5 years and an $R^2$ score of 0.81.
- **ResNet150V2 + EfficientNetB3 (Deep Features Only):** To assess the contribution of LBP features, this experiment focused solely on fusing deep features from the same CNN backbones. Performance improved, achieving an MAE of 4.9 years and an $R^2$ of 0.84.
- **VGG16 + EfficientNetB3:** A different combination, pairing VGG16 with EfficientNetB3, resulted in a comparatively higher MAE of 6.3 years and an $R^2$ score of 0.78.

The outcomes of these preliminary investigations suggested that fusing deep features learned by robust, pre-trained CNNs was more efficacious for this task than incorporating LBP features in conjunction with them. Furthermore, it was observed that deeper and more contemporary architectures, such as variants of ResNet, demonstrated greater promise. These insights guided our decision to explore the fusion of ResNet152V2 and DenseNet121 for our final proposed model.

### D. Proposed Model Architecture: ResNet152V2 and DenseNet121 Fusion

Our final model architecture is designed to harness the complementary feature extraction capabilities of two powerful deep CNNs: ResNet152V2 [8] and DenseNet121 [9]. Both networks were utilized with weights pre-trained on the large-scale ImageNet dataset [10], enabling effective transfer learning. The architecture consists of the following components:

- **Input Layer:** Accepts an RGB image of dimensions $224 \times 224 \times 3$.
- **Parallel Feature Extractors (Backbones):**
  - **ResNet152V2:** The ResNet152V2 model is employed without its original top classification layer (`include_top=False`). A Global Average Pooling (GAP) layer is appended to its output, transforming the feature maps into a 2048-dimensional feature vector, denoted as $f_{\text{ResNet}} \in \mathbb{R}^{2048}$.
  - **DenseNet121:** Similarly, the DenseNet121 model is used without its top classification layer. A GAP layer is applied to its output, resulting in a 1024-dimensional feature vector, denoted as $f_{\text{DenseNet}} \in \mathbb{R}^{1024}$.

ResNet architectures are renowned for their ability to train very deep networks effectively through the use of residual connections, while DenseNets promote extensive feature reuse and an improved flow of information and gradients throughout the network.

- **Feature Fusion Layer:** The feature vectors extracted from the two backbones, $f_{\text{ResNet}}$ and $f_{\text{DenseNet}}$, are concatenated to form a single, unified feature vector: $f_{\text{fused}} = [f_{\text{ResNet}}; f_{\text{DenseNet}}]$. This fused vector, $f_{\text{fused}} \in \mathbb{R}^{3072}$, captures a richer and more diverse set of features.
- **Regression Head:** The 3072-dimensional fused feature vector is then fed into a dedicated regression head, composed of a sequence of fully connected (Dense) layers:
  - Dense layer with 512 neurons and ReLU activation.
  - Dropout layer with a rate of 0.4 for regularization, mitigating overfitting.
  - Dense layer with 256 neurons and ReLU activation.
  - Dropout layer with a rate of 0.4.
  - An output Dense layer with a single neuron and a linear activation function, which directly predicts the continuous age value.

The total number of parameters in this fusion model is approximately 67 million, the majority of which are trainable during the fine-tuning process.

### *E. Training Details*

The proposed fusion model was trained using the following configuration and hyperparameters:

- **Optimizer:** The Adam optimizer [20] was employed, with an initial learning rate set to $1 \times 10^{-4}$. Adam is well-suited for training deep neural networks due to its adaptive learning rate capabilities.
- **Loss Function:** Mean Absolute Error (MAE) was chosen as the loss function. MAE directly penalizes the average absolute difference between the predicted ages and the true ground-truth ages, making it a suitable choice for age regression tasks.
- **Evaluation Metrics:** MAE was the primary metric used for monitoring performance during training and for model selection.
- **Callbacks for Training Control:**
  - `ModelCheckpoint`: This callback was configured to save the model weights that achieved the lowest MAE on the validation set during the training process.
  - `EarlyStopping`: To prevent overfitting and reduce unnecessary training time, EarlyStopping was implemented. It monitored the validation MAE and halted training if no improvement was observed for 10 consecutive epochs. The weights of the best performing model (based on validation MAE) were then restored.
  - `ReduceLROnPlateau`: This callback dynamically adjusted the learning rate. If the validation MAE did not show improvement for 5 consecutive epochs, the learning rate was reduced by a factor of 0.2.
- **Data Generation and Batching:** A custom data generator, inheriting from `tf.keras.utils.Sequence`, was developed for efficient loading and preprocessing of data in batches. A batch size of 16 was used.
- **Number of Epochs:** The model was set to train for a maximum of 50 epochs, with the EarlyStopping callback potentially terminating training sooner.

## IV. Experiments and Results

This section details the experimental setup, evaluation metrics, and the performance of our proposed age estimation model on the UTKFace and FG-NET datasets.

### *A. Evaluation Metrics*

To quantitatively assess the performance of our age estimation model, we employed the following standard metrics, widely used in the age estimation literature:

- **Mean Absolute Error (MAE):** This is the primary metric for regression-based age estimation. It is defined as the average absolute difference between the predicted age ($\hat{y}_i$) and the ground-truth age ($y_i$) for $N$ test samples:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{1}$$

  A lower MAE indicates better performance.
- **R-squared ($R^2$) Score:** Also known as the coefficient of determination, the $R^2$ score measures the proportion of the variance in the dependent variable (actual age) that is predictable from the independent variables (image features). It ranges from $-\infty$ to 1, where 1 indicates a perfect fit.
- **Cumulative Score (CS($\epsilon$)):** This metric calculates the percentage of test images for which the absolute prediction error, $|y_i - \hat{y}_i|$, is less than or equal to a predefined error tolerance level $\epsilon$ (in years). We report CS values for $\epsilon = \{1, 2, 3, 5, 7, 10\}$ years, where applicable, to provide a more nuanced understanding of the error distribution.

### *B. Performance on UTKFace Dataset*

The proposed fusion model was initially trained and evaluated on the UTKFace dataset. The training process was terminated by the EarlyStopping callback at epoch 25, with the model weights from epoch 15 (which yielded the best validation MAE of 4.8565 years) being restored for final evaluation. The performance metrics on the UTKFace test set are summarized in Table II.

TABLE II
Performance of the Proposed Model on the UTKFace Test Set

| Metric | Value |
|---|---|
| MAE (years) | **4.7076** |
| $R^2$ Score | **0.8871** |
| CS($\epsilon \leq 3$ years) (%) | 48.14 |
| CS($\epsilon \leq 5$ years) (%) | 66.27 |
| CS($\epsilon \leq 10$ years) (%) | 87.47 |

The model achieved an MAE of 4.71 years, indicating that, on average, its age predictions deviate from the actual age by this amount. The $R^2$ score of 0.8871 signifies a strong correlation between the predicted and actual ages, suggesting that the model explains a substantial portion of the variance in the age data. Furthermore, the CS scores are encouraging: 66.27% of the predictions fall within an error margin of $\pm 5$ years of the true age, and 87.47% within $\pm 10$ years.

### C. Qualitative Analysis on UTKFace

To gain deeper insights into the model's behavior, several qualitative analyses were performed based on its performance on the UTKFace dataset.

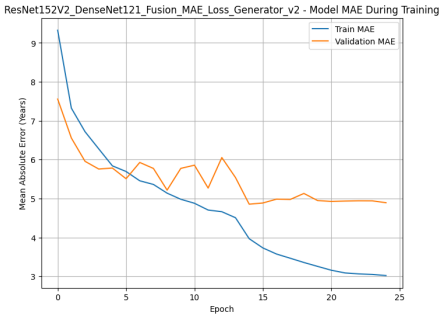


Fig. 1. Training and validation MAE curves over epochs for the proposed ResNet152V2-DenseNet121 fusion model on the UTKFace dataset. Early stopping occurred at epoch 25, restoring weights from epoch 15.

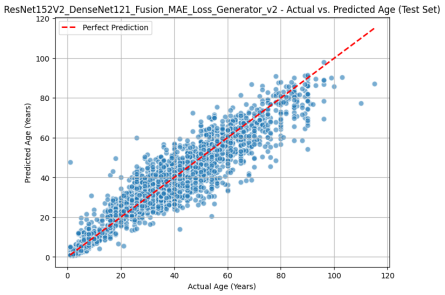


Fig. 2. Scatter plot of predicted age versus actual age on the UTKFace test set. The dashed red line represents perfect prediction ($y = x$).

Figure 1 depicts the MAE for both training and validation sets over the epochs. The training MAE consistently decreases, while the validation MAE plateaus around epoch 15, indicating that the EarlyStopping mechanism effectively prevented overfitting by restoring the model to its optimal generalization point. Figure 2 presents a scatter plot of predicted ages against actual ages. A strong positive correlation is evident, with most data points clustering around the ideal $y = x$ line, although a slight tendency to underestimate older ages and increased variance at extreme age ranges can be observed. The distribution of prediction errors, shown in Figure 3, is approximately Gaussian and centered close to zero (mean error of 0.70 years), suggesting no significant systematic bias in the predictions. Finally, the CS curve in Figure 4 visually confirms the model's accuracy across various error tolerances, reinforcing the CS values reported in Table II.

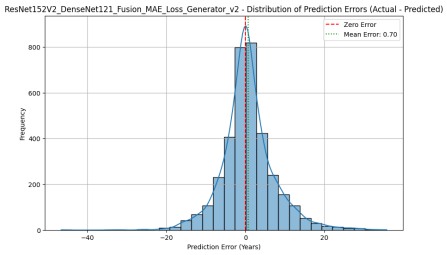


Fig. 3. Histogram illustrating the distribution of prediction errors (Actual Age - Predicted Age) on the UTKFace test set. The distribution is centered near zero with a mean error of 0.70 years.

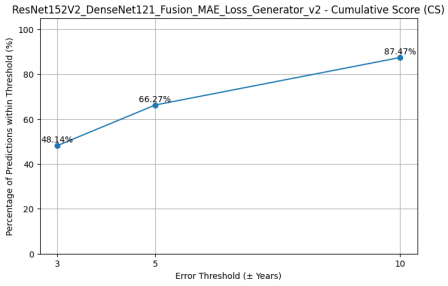


Fig. 4. Cumulative Score (CS) curve on the UTKFace test set, showing the percentage of predictions within a given error threshold ($\pm\epsilon$ years).

### D. Performance on FG-NET Dataset (Generalization Assessment)

To evaluate the generalization capability of our best model (trained on UTKFace), it was directly tested on the FG-NET dataset without any further fine-tuning. The performance results are presented in Table III.

When evaluated on FG-NET, the MAE increased modestly to 5.19 years, and the $R^2$ score decreased to 0.8222. This slight degradation in performance is anticipated when a model encounters a domain shift (i.e., a dataset with different characteristics from its training data). FG-NET is known for its specific challenges, including a smaller number of subjects, significant age variations per subject, and variable image quality. Despite these factors, an MAE of 5.19 years is a competitive result on FG-NET, and the CS scores, particularly

TABLE III
PERFORMANCE OF THE PROPOSED MODEL (TRAINED ON UTKFACE) ON THE FG-NET DATASET

| Metric | Value |
|---|---|
| MAE (years) | **5.1895** |
| $R^2$ Score | **0.8222** |
| CS($\epsilon \leq 1$ year) (%) | 18.06 |
| CS($\epsilon \leq 3$ years) (%) | 47.22 |
| CS($\epsilon \leq 5$ years) (%) | 63.19 |
| CS($\epsilon \leq 7$ years) (%) | 75.69 |
| CS($\epsilon \leq 10$ years) (%)— 82.72 | |

CS($\epsilon \leq 5$ years) at 63.19%, remain robust, underscoring the model's commendable generalization capabilities.

## V. CONCLUSION AND FUTURE WORK

In this paper, we have presented and comprehensively evaluated a high-performance system for facial age estimation, predicated on the synergistic fusion of two pre-trained deep convolutional neural networks: ResNet152V2 and DenseNet121. Our methodology emphasized meticulous data preprocessing, incorporating MTCNN-based face detection and alignment for input normalization, and an oversampling strategy to effectively address the common challenge of age distribution imbalance in training datasets. The proposed fusion model demonstrated strong predictive capabilities, achieving a Mean Absolute Error (MAE) of 4.71 years and an $R^2$ score of 0.8871 on the challenging UTKFace dataset. This performance not only surpasses that of several contemporary methods evaluated on the same dataset but also represents a significant improvement over prior institutional benchmarks. Furthermore, when evaluated on the FG-NET dataset, our model yielded an MAE of 5.19 years, indicating commendable generalization to unseen data with different characteristics.

The success of our approach robustly highlights the tangible benefits of leveraging diverse and deep feature hierarchies extracted from multiple pre-trained networks through an effective feature fusion mechanism. The results affirm that combining the distinct architectural strengths of well-established CNNs can lead to a more powerful, robust, and accurate age estimation system.

Looking ahead, several avenues for future research could further enhance the capabilities of our system. The implementation of more sophisticated data augmentation techniques, specifically designed to simulate subtle and diverse age-related facial changes, could improve model robustness and generalization. Investigating the integration of an ordinal regression framework with our powerful fusion backbone might offer a way to combine the benefits of structured output prediction with rich feature extraction. Additionally, the incorporation of attention mechanisms could potentially enable the model to dynamically focus on the most salient age-discriminative facial regions, leading to more refined predictions. Finally, conducting more extensive cross-dataset evaluations on a wider array of age estimation benchmarks would provide a more holistic understanding of the model's generalization performance across varied domains.

In summary, the proposed ResNet152V2 and DenseNet121 fusion model provides a strong and effective solution for facial age estimation, establishing a solid foundation for future research endeavors and potential deployment in practical applications.

## REFERENCES


[1] Y. Fu, G. Guo, and T. S. Huang, "Age synthesis and estimation via faces: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 1955–1976, 2010.

[2] K. Ricanek and T. Tesafaye, "MORPH: A longitudinal image database of normal adult age-progression," in *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*. IEEE, 2006, pp. 341–345.

[3] G. Levi and T. Hassner, "Age and gender classification using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 34–42.

[4] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output cnn for age estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4920–4928.

[5] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[6] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001.

[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[9] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. IEEE, 2009, pp. 248–255.

[11] R. Ranjan, C. D. Castillo, and R. Chellappa, "LBP-Net: A Deep LBP Based CNN for Facial Attribute-Aware Age Estimation," *arXiv preprint arXiv:1703.02709*, 2017.

[12] LisanneH, "Age estimation in supermarkets using utkface dataset," Hugging Face Model Card, 2023. [Online]. Available: https://huggingface.co/LisanneH/AgeEstimation

[13] E. Dogan, "Age estimation using basic cnn regression on utkface," GitHub Repository, 2020. [Online]. Available: https://github.com/emredogan7/age-estimation

[14] Various Authors, "Benchmarking CNNs on UTKFace for Age Estimation," *European Chemical Bulletin*, 2023, referring to Section 3. Available at: https://www.eurchembull.com/uploads/paper/3b90623d58316a790469ab181187bebd.pdf.

[15] ——, "Consistent Rank Logits for Ordinal Regression. UTKFace Benchmark Results," HyperAI Benchmark, 2023, from "Unraveling the Age Estimation Puzzle" series/project. Specific URL for UTKFace benchmark if available.

[16] L. AFENAI and K. BOUBEKRI, "Estimation de l'âge à partir des images faciales par les réseaux de neurones convolutifs (cnn)," Mémoire de Master Recherche, Université Abderrahmane Mira de Bejaia, Bejaia, Algeria, 2022.

[17] Z. Zhang, Y. Song, and H. Qi, "Age progression/regression by conditional adversarial autoencoder," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5810–5818.

[18] A. Lanitis, C. J. Taylor, and T. F. Cootes, "Toward automatic simulation of aging effects on face images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 442–455, 2002.

[19] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

# References

[1] P. Kamencay, M. Benco, T. Mizdos, and R. Radil, "A new method for face recognition using convolutional neural network," *Advances in Electrical and Electronic Engineering*, vol. 15, no. 4, pp. 623–632, 2017.

[2] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[3] K. Kittusamy, B. Krishnakumar, A. S. Aswath, P. S. Gowtham, and S. R. Vishal, "Terrain identification and land price estimation using deep learning," in *AIP Conference Proceedings*, vol. 2387, pp. 140030–1–140030–5, AIP Publishing, 2021.

[4] H. Al-barazanchi, H. Qassim, and A. Verma, "Residual cnds," August 2016. arXiv preprint, CC BY 4.0 license.

[5] DataCorner.fr, "Image processing – partie 6." `https://datacorner.fr/image-processing-6/`, 2025. Consulté en juin 2025.

[6] Inside Machine Learning, "Fonction d'activation : Comment ça marche ? une explication simple." `https://inside-machinelearning.com/fonction-dactivation-comment-ca-marche-une-explication-simple/`, 2022. Consulté en juin 2025.

[7] T. Kalsum and Z. Mehmood, "A novel lightweight deep convolutional neural network model for human emotions recognition in diverse environments," *To be completed*, 2023. Department of Software Engineering and Computer Engineering, UET Taxila, Pakistan.

[8] A. Al and M. Z. Ami, "A brief explanation of convolutional neural network with practical implementation in keras & tensorflow." Online article, 2025. Machine Learning Deep Learning Researcher ; Co–Founder of Wavy AI Research Foundation.

[9] G. Jee, H. Gm, M. K. Gourisaria, M. Pandey, *et al.*, "Efficacy determination of various base networks in single shot detector for automatic mask localisation in a post covid setup," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 35, no. 3, 2022.

[10] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[11] Y. Fu, Y. Xu, and T. S. Huang, "Estimating human age by manifold analysis of face pictures and regression on aging features," in *International Conference on Multimedia and Expo (ICME)*, pp. 1383–1386, IEEE, 2007.

[12] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708, 2017.

[13] A. □. compléter, "An efficient face detection system using the viola-jones algorithm," *International Journal of Research Publication and Reviews*, vol. à compléter, no. à compléter, p. à compléter, 2023.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580–587, IEEE, 2014.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN : Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 28, 2015.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once : Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.

[17] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou, "Retinaface : Single-stage dense face localisation in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5203–5212, 2020.

[18] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition : A literature survey," *ACM Computing Surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[19] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[20] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *Proceedings of the 19th International Conference on Pattern Recognition (ICPR)*, pp. 1–4, IEEE, 2008.

[21] J. Kannala and E. Rahtu, "Bsif : Binarized statistical image features," *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*, pp. 1363–1366, 2012.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[23] M. Tan and Q. V. Le, "Efficientnet : Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97, pp. 6105–6114, PMLR, 2019.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR)*, 2015. arXiv :1409.1556.

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, pp. 1097–1105, Curran Associates, Inc., 2012.

[27] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning : with Applications in R*. Springer Texts in Statistics, New York : Springer, 1 ed., 2013.

[28] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Proceedings of the 8th European Conference on Computer Vision (ECCV)*, vol. 3021 of *Lecture Notes in Computer Science*, pp. 469–481, Springer, 2004.

[29] A. Gunay and V. V. Nabiyev, "Automatic age classification with lbp," in *20th International Conference on Pattern Recognition (ICPR)*, pp. 3396–3399, IEEE, 2010.

[30] S. E. Bekhouche, A. Ouafi, A. Taleb-Ahmed, A. Hadid, and A. Benlamoudi, "Facial age estimation using bsif and lbp," in *International Conference on Image Processing Theory, Tools and Applications (IPTA)*, IEEE, 2017.

[31] V. Ojansivu and J. Heikkilä, "Methods for local phase quantization in blur-insensitive image analysis," *Signal Processing : Image Communication*, vol. 27, no. 6, pp. 550–561, 2012.

[32] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[33] R. Cament, F. Galdames, K. W. Bowyer, and C. A. Perez, "Face recognition under pose variation with local gabor features enhanced by active shape and statistical models," *Pattern Recognition*, vol. 48, no. 11, pp. 3257–3271, 2015.

[34] P. Dey, T. Mahmud, M. S. Chowdhury, M. S. Hossain, and K. Andersson, "Human age and gender prediction from facial images using deep learning methods," *Procedia Computer Science*, vol. 222, pp. 1234–1241, 2024.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[36] I. Aruleba and S. Viriri, "Deep learning for age estimation using efficientnet," in *Advances in Computational Intelligence : 16th International Work-Conference on Artificial Neural Networks (IWANN 2021)*, vol. 12861 of *Lecture Notes in Computer Science*, pp. 407–419, Springer, 2021.

[37] R. Thaneeshan, K. Thanikasalam, and A. Pinidiyaarachchi, "Gender and age estimation from facial images using deep learning," in *2022 7th International Conference on Information Technology Research (ICITR)*, pp. 1–6, IEEE, 2022.

[38] S. Zaghbani, N. Boujneh, and M. S. Bouhlel, "Age estimation using deep learning," *Computers & Electrical Engineering*, vol. 68, pp. 337–347, 2018.

[39] N. Mualla, E. H. Houssein, and H. H. Zayed, "Face age estimation approach based on deep learning and principal component analysis," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 2, pp. 152–158, 2018.

[40] G. Castellano, B. D. Carolis, N. Marvulli, M. Sciancalepore, and G. Vessio, "Real-time age estimation from facial images using yolo and efficientnet," in *Proceedings of the International Conference on Image Analysis and Processing (ICIAP)*, (Bari, Italy), Springer, 2023.

[41] P. Smith and C. Chen, "Transfer learning with deep cnns for gender recognition and age estimation," in *Proceedings of the IEEE International Conference on Big Data (Big Data)*, pp. 2423–2432, IEEE, 2018.

[42] S. Chen, C. Zhang, and M. Dong, "Deep age estimation : From classification to ranking," *IEEE Transactions on Multimedia*, vol. PP, no. 99, pp. 1–1, 2017.

[43] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output cnn for age estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4920–4928, IEEE, 2016.

[44] Z. Hu, Y. Wen, J. Wang, M. Wang, R. Hong, and S. Yan, "Facial age estimation with age difference," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, IEEE, 2016.

[45] K. Li, J. Xing, W. Hu, and S. J. Maybank, "D2c : Deep cumulatively and comparatively learning for human age estimation," *Pattern Recognition*, vol. 98, p. 107078, 2020.

[46] A. A. Micheal and R. Shankar, "Automatic age and gender estimation using deep learning and extreme learning machine," *Procedia Computer Science*, vol. 171, pp. 2373–2380, 2020.

[47] A. Younesi, M. Ansari, M. Fazli, A. Ejlali, M. Shafique, and J. Henkel, "A comprehensive survey of convolutions in deep learning : Applications, challenges, and future trends," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[48] S. Ioffe and C. Szegedy, "Batch normalization : Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pp. 448–456, PMLR, 2015.

[49] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML)*, (Haifa, Israel), pp. 807–814, 2010.

[50] Y. H. Sabri, W. Z. W. Hasan, A. H. Sabry, S. Shafie, *et al.*, "Measurement-based modeling of a semitransparent cdte thin-film pv module based on a custom neural network," *IEEE Access*, vol. 6, pp. 1–1, June 2018. CC BY-NC-ND 4.0 License.

[51] S. S. Li, "Utkface dataset." `https://susanqq.github.io/UTKFace/`. Accessed June 2025.

[52] A. Lanitis, M. Draganis, C. Christodoulou, and N. Tsapatsoulis, "An overview of research on facial aging using the fg-net aging database," *IET Biometrics*, vol. 4, pp. 219–226, May 2015.

[53] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output cnn for age estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[54] LisanneH, "Age estimation in supermarkets using utkface dataset." `https://huggingface.co/LisanneH/AgeEstimation`, 2023. Hugging Face Model Card.

[55] E. Dogan, "Age estimation using basic cnn regression on utkface." `https://github.com/emredogan7/age-estimation`, 2020. GitHub Repository.

[56] Auteurs Divers, "Benchmarking cnns on utkface for age estimation," *European Chemical Bulletin*, 2023.

[57] Auteurs Divers, "Consistent rank logits for ordinal regression," 2023. Unraveling the Age Estimation Puzzle - HyperAI Benchmark.

[58] L. AFENAI and K. BOUBEKRI, "Estimation de l'âge à partir des images faciales par les réseaux de neurones convolutifs (cnn)," mémoire de master recherche, Université Abderrahmane Mira de Béjaïa, Algérie, 2022. Présenté dans le cadre d'un Master en Informatique, spécialité Intelligence Artificielle.

# Abstract

Age estimation from facial images is a significant challenge in the field of computer vision, with diverse applications ranging from security to demographic analysis and human-machine interfaces. In this context, this thesis proposes an effective approach combining powerful CNN architectures and direct regression, demonstrating competitive performance compared to existing methods.

Our system follows a complete pipeline comprising several key steps : face detection using MTCNN, alignment, resizing, followed by feature extraction and fusion of characteristics derived from two CNN architectures, namely ResNet152V2 and DenseNet121. These features are then fused and used as input to a direct regression model, enabling age prediction as a continuous numerical value.

The model was trained and evaluated on the UTKFace dataset, and subsequently tested on FG-NET to assess its generalization capability. The results obtained show a Mean Absolute Error (MAE) of 4.71 years on UTKFace and 5.19 years on FG-NET.



# Résumé

L'estimation d'âge à partir d'images faciales constitue un défi important dans le domaine de la vision par ordinateur, avec des applications variées telles que la sécurité, l'analyse démographique et les interfaces homme-machine. Dans ce contexte, ce mémoire propose une approche efficace combinant des architectures CNN puissantes et une régression directe, démontrant une performance compétitive par rapport aux méthodes existantes.

notre système suit un pipeline complet comprenant plusieurs étapes clés : la détection du visage à l'aide de MTCNN , une étape d'alignement , de redimensionnement, suivie de l'extraction et de la fusion des caractéristiques issues des deux réseaux CNN,à savoir ResNet152V2 et DenseNet121.Ces caractéristiques sont ensuite fusionnées et utilisées comme entrée d'un modèle de régression directe, permettant de prédire l'âge sous forme d'une valeur numérique continu.

Le modèle a été entraîné et évalué sur la base de données UTKFace , puis testé sur FG-NET afin d'évaluer sa capacité de généralisation. Les résultats obtenus montrent une erreur absolue moyenne (MAE) de 4,71 ans sur UTKFace et de 5,19 ans sur FG-NET.