Estimation of people's age from facial photos using Deep Learning

Javier Tomás Fernández Martín

Abstract

This work employs deep learning techniques to tackle the challenge of age estimation from facial images. The project explores and implements state-of-the-art convolutional neural networks (CNNs), evaluating multiple architectures, including MobileNetV2, ResNet50, EfficientNetB4, and EfficientNetB5, to develop a robust and scalable solution for age classification. Given the complexities of age variation—such as lighting, pose, and expression—it was necessary to incorporate data augmentation strategies and mitigate class imbalances through targeted oversampling and weighted loss functions. To simplify classification and enhance accuracy, age groups were organized into predefined ranges. Transfer learning and finetuning of pre-trained models allowed effective adaptation to limited datasets, significantly improving performance. The evaluation extends beyond accuracy, incorporating Mean Absolute Error (MAE), balanced accuracy, Cohen's Kappa, and confusion matrices to analyze model behavior comprehensively. Results demonstrate that EfficientNetB4 achieved the highest test accuracy of 65.87%, with an MAE of 0.4723, while EfficientNetB5 provided better class balance but at the cost of lower overall accuracy (60.43%). The findings highlight the trade-offs between model complexity and performance, emphasizing the impact of architecture selection and data balancing strategies. This work provides a foundation for practical applications in areas such as security, marketing, and healthcare while identifying future research directions, including alternative feature extraction techniques, the exploration of generative models, and improvements in dataset construction to enhance the robustness of age estimation.

Index Terms

Age estimation, deep learning, convolutional neural networks (CNNs), EfficientNetB4, transfer learning, data augmentation, class imbalance, facial analysis, and computer vision.

I. Introduction

AGE estimation from facial images is a challenging task within the field of computer vision. Its automatic implementation has applications in numerous domains, including security, marketing, health diagnostics, and human-computer interaction [44, 17]. However, accurately estimating age is difficult even for humans due to significant variations caused by genetics, lifestyle, environmental factors [39, 9] and image conditions such as lighting, pose, and occlusions [11, 16]. This work addresses the challenge of age estimation using **deep learning models** based on **convolutional neural networks (CNNs)**. Specifically, it focuses on developing a robust and scalable approach by leveraging various pre-trained neural networks, with **EfficientNetB4** as a key component. This architecture stands out as a state-of-the-art model, renowned for its balance between computational efficiency and accuracy. By training the model on datasets segmented into 10-year and 5-year age ranges, the proposed approach aims to improve generalization across diverse demographics and evaluate the impact of different segmentations on performance.

Despite recent advancements in deep learning, key challenges persist:

1) **Class imbalance** - Older age groups tend to have fewer examples, leading to biased predictions
2) **Overfitting** - High-dimensional deep learning models are prone to overfitting, particularly with limited training data.
3) **Evaluation metrics** - Traditional accuracy metrics may not fully capture the performance of models in age estimation tasks, where predictions close to the actual age might still be considered acceptable.

To address these challenges, this research proposes solutions such as:

1) **Data augmentation** - Strategies targeted at minority classes to balance datasets [35, 40].
2) **Dynamic learning rates** - Adaptive optimizers to improve convergence and generalization [36, 25].
3) **Confusion matrix and error-based metrics** - Metrics such as **Mean Absolute Error (MAE)** to complement accuracy measurements [7, 42].

A. Motivation and Hypothesis

The motivation for this work stems from the increasing demand for accurate age estimation systems in practical applications, such as:

1) **Surveillance and security** - Estimating age to help identify underage individuals or potential threats [1, 16].
2) **Healthcare** - Enabling age-based screening and analysis of age-related diseases [39, 15].
3) **Marketing and retail** - Facilitating targeted advertising and personalized recommendations [17, 32].

Author: Javier Tomás Fernández Martín
Advisor: Manuel Ujaldón
Thesis dissertation submitted: February 2025

The hypothesis is that leveraging pre-trained architectures like **EfficientNetB4**, combined with data augmentation and balanced sampling techniques, can significantly enhance model performance. Specifically, this approach is expected to achieve:

1) Improved accuracy and generalization across diverse age groups.
2) Better performance under challenging conditions such as occlusions and poor lighting.
3) Competitive results when compared with existing state-of-the-art methods.

This paper is organized as follows:

- Section II provides a comprehensive review of related works.
- Section III details the methodology and design of the proposed solution.
- Section IV presents experimental results and analysis.
- Section V discusses the conclusions and future research directions.

II. State of the Art

The problem of age estimation from facial images has attracted significant research attention due to its applications in fields such as security, healthcare, and human-computer interaction. Traditional approaches relied on handcrafted features, which often lacked robustness and accuracy when dealing with variations in pose, lighting, and occlusions, as well as the inevitable human error in this field [3]. Recent advances in **deep learning** have revolutionized this area by enabling automatic feature extraction and representation learning, significantly improving performance [2].

Moreover, the use of CNNs has facilitated the transition from manually designed features to automatically learned representations, addressing many challenges inherent to image-based problems compared to tabular data.

A. Early Methods

Early researchs have been performed with techniques like **Local Binary Patterns (LBP)**, **Gabor filters**, and **Active Appearance Models (AAM)** for feature extraction [2]. These methods were intended to detect characteristics like texture, shape, and wrinkles which are important in the age estimation process. However, their sensitivity to differences in image quality, illumination, and facial expressions resulted in inconsistent outputs [21].

Traditional machine learning models, such as support vector machines (SVMs) and linear regressors, have been employed for age estimation tasks, either by classifying individuals into predefined age categories or by performing continuous age regression. For instance, a study utilizing SVMs for human age estimation from facial images demonstrated the application of these models in this domain [45, 10].

However, these traditional models often face limitations when addressing the complexities of age estimation. Linear regression models assume a linear relationship between variables, which may not capture the nonlinear patterns present in aging processes. Additionally, they are sensitive to outliers, potentially skewing predictions [23].

Similarly, while SVMs can handle nonlinear relationships through kernel functions, they may still struggle with highdimensional data and complex variations in facial features associated with aging. Moreover, both SVMs and linear regressors may not adequately address issues such as class imbalance and the subtlety of age-related changes, leading to less accurate predictions compared to more advanced methods [13].

These restrictions suggest the need to employ more sophisticated strategies, e.g. deep learning models, which are capable of learning complex relationships and features typical for age estimation problems.

B. Neural Networks

1) Artificial Neural Networks (ANNs): ANNs are computational models that are designed based on the idea of biological neural networks [27]. An ANN consists of layers of interconnected nodes, or neurons, that process information by applying weights and biases to input data. These weights are adjusted during training to minimize the error between the predicted output and the ground truth [31].

The three primary types of layers that composed an ANNs are:

1) **Input Layer** - Input for the raw data.
2) **Hidden Layers** - Extract and learn hierarchical features through nonlinear transformations.
3) **Output Layer** - Gives the final prediction or classification result.

Training an ANN involves optimizing a loss function using techniques such as backpropagation and gradient descent [33]. The network learns complex patterns in the data by adjusting its parameters iteratively.

2) Convolutional Neural Networks (CNNs): CNNs are a type of ANNs that are designed specifically for working with data that is presented in a grid format, particularly images [20]. They contain convolutional layers that apply filters on the data to identify spatial features, pooling layers for reducing the dimensionality of the input, and fully connected layers for the last classification. This architecture is very successful in computer vision tasks due to its ability to learn both local and global features of the input.

Key advantages of CNNs include:

1) Computational cost is reduced through the use of parameter sharing.
2) Robustness to variations in input data, such as translations and distortions.
3) Automatic feature extraction, eliminating the need for manual engineering [19].

The advent of CNNs revolutionized the field, enabling significant advancements in tasks like image classification, object detection, and age estimation. Early models such as **VGG-Face** [30] and **AgeNet** [21] demonstrated the potential of CNNs to handle complex visual data, laying the groundwork for modern architectures.

C. Transfer Learning

Transfer learning is a deep learning approach where a pre-trained model is fine-tuned for a different but related task using a large initial dataset [29]. This approach leverages the hierarchical feature representations learned during pre-training and achieves significant improvement in terms of training time and performance, especially for tasks with limited data [43].

The primary advantages of transfer learning include:

1) **Reduced Computational Resources** - Pre-trained models save the need for huge training on large-scale datasets.
2) **Improved Generalization** - Features learned from diverse datasets allows better adaptation to new tasks.
3) **Faster Convergence** - Fine-tuning requires less epochs compared to training a model from scratch.

In this work, transfer learning has been used with several state-of-the-art pre-trained architectures, explained below:

1) ResNet-AffectNet: ResNet (Residual Networks) introduced skip connections to address the vanishing gradient problem, allowing for the successful training of very deep networks [18]. The ResNet-AffectNet variant is fine-tuned on the AffectNet dataset, specializing in facial emotion and attribute recognition [28]. Its robust feature extraction capabilities make it well-suited for age estimation tasks.

2) EfficientNetB4: EfficientNetB4 is part of the EfficientNet family, which scales model depth, width, and resolution systematically using a compound scaling method [37]. This architecture achieves state-of-the-art results while maintaining computational efficiency, making it ideal for tasks requiring a balance between accuracy and resource constraints.

3) MobileNetV2: MobileNetV2 is a lightweight architecture designed for mobile and embedded devices [34]. It employs inverted residuals and linear bottlenecks, achieving high performance with reduced computational cost. Its efficient design makes it suitable for scenarios where computational resources are limited.

4) ResNet50: ResNet50 is a widely used variant of the ResNet architecture with 50 layers [18]. Its deep architecture enables the extraction of detailed and hierarchical features, making it a popular choice for transfer learning in a variety of vision tasks.

D. Modern Architectures

Recent advancements focus on improving the robustness of models under real-world conditions. For instance **Low-Complexity Attention Generative Adversarial Networks (LCA-GANs)** address challenges like occlusions by enhancing feature extraction from critical facial regions, including the eyes, nose, and mouth [24]. Attention mechanisms have also been incorporated into the architectures to learn the age relevant regions focused attention to improve the performance in pose, illumination and noise variation [2].

Furthermore, models that integrate super resolution techniques such as GAN based models improve low quality images and recover age relevant details like wrinkles and texture [2].

E. Differences Between Tabular and Image Data

Facial age estimation is quite different from issues with tabular data, which often rely on structured and clearly defined features. Conversely, images are structured data and need to through special preprocessing and feature extraction methods. CNNs overcome this field by learning hierarchically the spatial and temporal relationships of an image's pixels. More- rather, image models will have to contend with high-dimensionality and demand strong architectures and augmentation techniques for effective generalizing

F. Advances in Facial Landmark Detection

Facial landmarks are specific points on a human face that can be used to represent facial features, such as the eyes, nose, mouth, and the overall face outline. These points are typically detected through computational models, enabling the alignment and recognition of facial structures in images. Landmark detection has become a critical tool in computer vision tasks, including facial recognition, emotion detection, and age estimation [45].

In the context of age estimation, facial landmarks play a crucial role as they help identify the key structural features that change with age. For example, the shape and position of facial landmarks, such as the eyes, eyebrows, and mouth, evolve over time, providing valuable information for accurate age prediction. Moreover, landmarks can enhance the robustness of age estimation models by normalizing facial images for variations in pose, scale, and lighting [8].

Landmark detection techniques have been used for years, and the technology to obtain them has evolved significantly, especially in recent times due to the rapid advancements in deep learning and CNNs. This evolution can be understood by examining the progression from traditional approaches, through the emergence of modern methods, to the development of robust landmark detectors like dlib and Mediapipe, which are widely used today.

1) ***Traditional Approaches****:* Early methods relied on handcrafted features such as Active Appearance Models (AAMs) and Active Shape Models (ASMs), which used predefined face templates to locate key facial points. While these techniques were groundbreaking at the time, they were often hindered by challenges such as variations in orientation, environmental conditions, and partial obstructions of the face [14].

2) ***Modern Methods****:* With the recent advancements of deep learning, landmark detection has become more precise and robust. Tools such as dlib introduced a 68-point facial landmark detector, as shown in Figure 1, using ensemble regression trees, which became a standard for many facial analysis tasks. However, despite its efficiency, the limited number of landmarks offered by dlib constrains its application in tasks requiring finer details, such as age estimation and emotion recognition [21]. Additionally, classical techniques like Haar cascades were once widely used for face detection but have been largely surpassed by modern deep learning-based approaches due to their lower accuracy and robustness in diverse conditions [38].

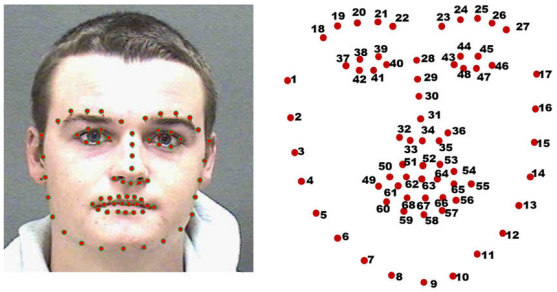

Fig. 1: Face mesh using Dlib

Recent advancements, such as MediaPipe FaceMesh, have redefined the state of the art by offering 468 high-density facial landmarks, as shown in Figure 2. This increased number of points allows for more detailed segmentation of facial regions, capturing subtle features critical for tasks like age estimation. MediaPipe also excels in real-time applications, offering fast and robust performance even under challenging conditions, such as variable lighting or partial occlusions [24]. Its high density and efficiency have made it a popular choice for applications in age and gender classification, where precise facial feature extraction is essential [22].

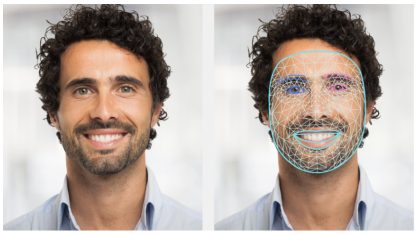

Fig. 2: Face landmarks detected by MediaPipe

These advancements in landmark detection technologies have significantly enhanced the accuracy and applicability of facial analysis systems, particularly in real-time applications and in environments with limited computational resources. These improvements have paved the way for more precise tasks, such as the challenge discussed in this work.

G. Challenges and Improvements

Despite the progress, age estimation models still face several challenges:

- **Pose and Illumination Variations:** Augmentation techniques, such as rotation, flipping, cropping, and brightness adjustments, are essential for mitigating these issues [12].
- **Low-Resolution Images:** Super-resolution methods, including GAN-based reconstructions, enhance low-quality inputs, ensuring critical facial features are preserved [2].
- **Dataset Limitations:** Popular datasets like **UTKFace**, **FG-NET**, and **MORPH** often lack demographic diversity and balanced age group distributions, leading to biases in model predictions [26].

H. Evaluation Metrics

To evaluate the performance of age estimation models, several metrics are commonly used, including:

a) Mean Absolute Error (MAE): MAE quantifies the average absolute difference between the predicted class and the true class. This metric provides a direct and interpretable measure of the model's performance, as it reflects the magnitude of the errors without being influenced by outliers. MAE is particularly useful in age estimation because it emphasizes minimizing the error in each individual prediction, ensuring the model's output is as close as possible to the true age value [41].

b) Cumulative Score (CS): CS measures the accumulated prediction accuracy across multiple age intervals. In age estimation, it is important to place individuals in the correct age range. CS evaluates how well the model can categorize individuals into the appropriate age ranges, making it valuable for tasks that require grouping predictions into broader age categories [5].

c) Confusion Matrices: Confusion matrices are commonly used in classification tasks and can be adapted for regression problems by treating age ranges as categories. The confusion matrix shows how many instances fall into each predicted age range versus the true age range, allowing for an evaluation of the model's ability to generalize across different age groups. This metric is particularly useful for identifying where the model performs well and where it needs improvement, as it indicates which classes the model confuses with one another. [4].

d) Why Accuracy Is Not Enough?: In contrast to classification tasks where accuracy is a common evaluation metric, age estimation models are often better evaluated using MAE or CS. Accuracy in age estimation can be misleading because predicting an exact age (e.g., 23 years) is much more difficult than predicting an age range (e.g., between 20 and 30 years).This issue doesn't exists if the target are age ranges, but accuracy does not reflect how close or far the predictions are from the true values, which is particularly critical in age estimation tasks. For instance, predicting 20 years for a 30-year-old is different from predicting 20 years for a 70-year-old. MAE provides a more nuanced understanding of model performance by highlighting the magnitude of error in predictions [6].

I. Conclusion

The evolution from traditional handcrafted features to deep learning-based approaches has significantly advanced the field of facial landmark detection and age estimation. Techniques like transfer learning and dense landmark detectors, such as **MediaPipe**, have enhanced both accuracy and robustness, enabling more precise and real-time applications. Despite these advancements, challenges remain in areas like dataset diversity, pose variation, and low-resolution images. This work builds upon the latest deep learning innovations, utilizing advanced architectures and augmentation methods to propose a scalable and effective solution for age estimation. The approach outlined in this study aims to address these challenges while offering a pathway to further improvements in real-world applications.

III. Method

This section describes the computational approach used to solve the problem of age estimation. It includes details about data organization, preprocessing, and dataset preparation.

A. Datasets Used

This section describes the datasets used in the age estimation process, highlighting their characteristics, image distribution, total number of samples, and the naming conventions used for class identification.

1) UTKFace Dataset: Due to its size and diversity, the UTKFace dataset is one of the most widely used databases for age estimation tasks. It contains facial images labeled with information about age, gender, and ethnicity, making it suitable for a variety of computer vision applications.

- **Number of images:** 23,708 facial images.
- **Age range:** From 0 to 116 years old.
- **Resolution:** Images have variable resolutions, typically ranging from low to medium quality.
- **Age distribution:** The dataset exhibits significant imbalance, with a higher concentration of images in younger age groups (0-30 years) and fewer images for older age groups (70 years and above).

- **Naming convention:** Each image is named following the format `age_gender_race.jpg`. For example:
  - `25_0_2.jpg`: This represents a 25-year-old individual, gender is male (`0`), and ethnicity is coded as `2`.
  - `50_1_1.jpg`: This represents a 50-year-old individual, gender is female (`1`), and ethnicity is coded as `1`.

2) FG-NET Dataset: The FG-NET dataset is another widely recognized database used for facial age estimation. Although smaller in size compared to UTKFace, it provides a valuable resource for evaluating age estimation methods.

- **Number of images:** 1,002 facial images.
- **Age range:** From 0 to 69 years old.
- **Image diversity:** This dataset includes multiple images of the same individual taken at different ages, providing a longitudinal perspective.
- **Age distribution:** The dataset is more evenly distributed across age groups compared to UTKFace, but there are fewer images for extreme age ranges (e.g., 60+ years).
- **Naming convention:** Images are named following a custom convention that encodes the individual's ID and age. For example:
  - `001A07.jpg`: This represents the first individual (`001`), with age 7 years (`07`).
  - `045B35.jpg`: This represents the 45th individual (`045`), with age 35 years (`35`).

3) Aging Faces in the Wild (AGFW) Dataset: The Aging Faces in the Wild (AGFW) dataset is a large-scale collection of facial images specifically designed to support age estimation tasks. It focuses on providing natural, in-the-wild facial images that encompass a wide variety of real-world conditions.

- **Number of images:** Approximately 18,685 facial images.
- **Age range:** From 0 to 100+ years old.
- **Image diversity:** The dataset is characterized by high variability in pose, lighting, and background conditions, simulating real-world scenarios. It also includes images with occlusions, different expressions, and varying quality.
- **Age distribution:** The dataset has a relatively balanced age distribution compared to others. However, there is still a higher concentration of images in younger age groups (0-30 years) and fewer samples for older individuals (70+ years).
- **Naming convention:** Images follow a structured naming format encoding the estimated age and other demographic attributes. For example:
  - `age_30_male.jpg`: Indicates an individual with an estimated age of 30 years, gender as male.
  - `age_55_female.jpg`: Indicates an individual with an estimated age of 55 years, gender as female.
- **Challenges:** The "in-the-wild" nature of the images introduces complexities such as non-frontal faces, varying resolutions, and occlusions (e.g., glasses, hats), which make it a more challenging dataset for age estimation tasks.

B. Data Organization and Preprocessing

1) Age-Based Organization of Datasets: The preprocessing stage focused on organizing and preparing the datasets to ensure their compatibility with the age estimation model. The main goal of this process was to consolidate all datasets into a unified structure, where each image would be placed in a corresponding subfolder named after its specific age. This structure facilitated the subsequent grouping of images into broader age-range folders. Since multiple datasets were used, each with distinct file naming conventions and formats, tailored preprocessing steps were applied to handle their unique characteristics.

a) FG-NET Dataset: Images from the FG-NET dataset were organized by extracting the age information from the last two digits of their file names. A script was developed to create directories for each age and systematically move the images into their corresponding folders. To prevent issues with duplicate file names when merging datasets, the script also renamed the images using a sequential numbering system (e.g., 1, 2, ...). Images with invalid naming conventions were excluded to avoid processing errors, ensuring consistency and reliability in the dataset structure throughout the organization process.

```
organizar_por_edad_FGNET(input_folder="path_to_FGNET/images",
                         output_base_folder="dataset")
```

b) UTKFace Dataset: The UTKFace dataset underwent a similar processing approach, where the age was extracted from the prefix of each file name (preceding the underscore). Following the procedure used for the FG-NET dataset, images with invalid naming conventions were excluded to maintain dataset integrity. Valid images were renamed sequentially to extend the numbering within each folder, ensuring unique file names and avoiding conflicts when merging datasets.

```
organizar_por_edad_UTKPlus(input_folder="path_to_UTKFace",
                           output_base_folder="dataset")
```

c) Reorganizing Age Folders into Ranges: After processing the FG-NET and UTKFace datasets, where images were extracted and organized based on their absolute age values, the next step involved standardizing the dataset structure. Since

these datasets were initially organized in folders representing individual ages (e.g., `0`, `1`, ..., `120`), a reorganization was necessary to align them with the age-range structure required for the model and the AGFW dataset.

A script was implemented to automate this process. It iterated through each folder representing an absolute age and assigned the images to new folders based on predefined ranges. These ranges were defined in intervals of five years as in the AGFW dataset (e.g., `age_0_4`, `age_5_9`, ..., `age_115_119`). For each image, the script:

- Determined the corresponding age range based on the folder name.
- Created the range folder if it did not already exist.
- Renamed the image following the unique sequential identifier (e.g., `1.jpg`, `2.jpg`, ...) to avoid conflicts during merging.

This reorganization step was essential for maintaining consistency across datasets and simplifying future integrations or transformations. It also ensured that images with invalid names or without a corresponding folder were excluded, preserving the integrity of the data.

```
copiar_a_rangos(input_folder="dataset",
                output_folder="dataset_ranges")
```

d) AGFW Dataset: The AGFW (Aging Faces in the Wild) dataset required a tailored preprocessing approach due to its unique structure. Images were initially organized into gender-based folders (”male” and ”female”), with each further subdivided into age ranges of five years (e.g., `age_0_4`, `age_5_9`, ...). A script was executed for each gender folder to merge all images into a unified structure based on these age ranges. During this process, valid image files were copied into destination folders corresponding to their respective age ranges, and any invalid or unsupported files were excluded.

Once again, renaming the images sequentially continuing the numbering from the existing folder was used to prevent filename conflicts during the merging process. This preprocessing step ensured that the AGFW dataset seamlessly integrated with the unified age-based structure used for the other datasets. By consolidating the AGFW dataset alongside FG-NET and UTKFace, a robust and well-organized dataset was created to train the age estimation model effectively.

```
organizar_por_edad_AGFW(origen_folder="path_to_AGFW/female",
                        destino_folder="dataset_ranges")
```

2) Validation of Data Organization: A validation script was used to count the total number of images and verify the distribution across all folders. This step ensured the integrity and completeness of the dataset creation for this challenge.

```
contar_imagenes(base_folder="dataset_ranges")
```

The script displayed the number of images in each folder and provided a total count to confirm that no data was lost during preprocessing.

All this preprocessing led to a dataset distributed as shown in Figure 3. Due to the significant imbalance in certain age ranges, this distribution might pose challenges for the model to generalize effectively, as it could become biased towards the more represented classes.

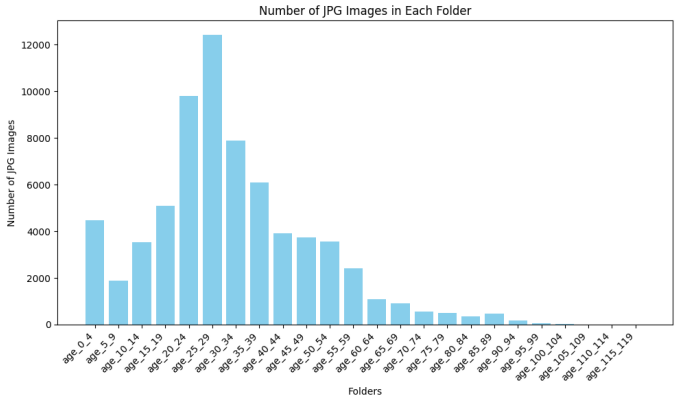

Fig. 3: Distribution of images across 5-year age ranges.

To mitigate this issue, broader age ranges of 10 years were created, resulting in the dataset distribution shown in Figure 4. While this adjustment does not fully resolve the imbalance problem, partially due to some corrupted images during the new organization and some image loss, it aimed to slightly smoothen it by grouping underrepresented age ranges into broader categories. Despite this, significant disparities remain, particularly in the higher age ranges, which could still pose challenges for model generalization.

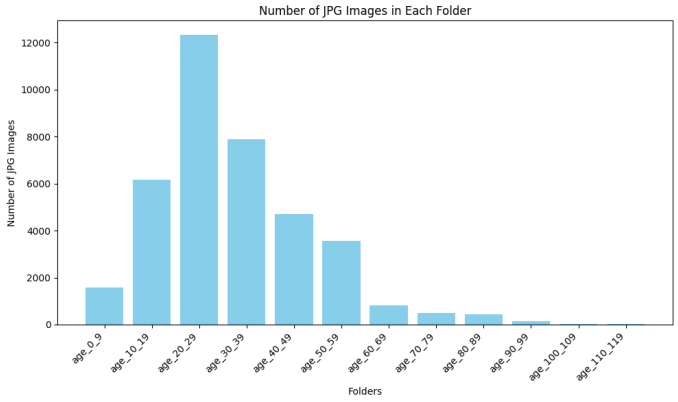

Fig. 4: Distribution of images across 10-year age ranges.

Throughout the project, both datasets were utilized for training and evaluation. This allowed for a comparative analysis of how the different age range groupings impact the model's performance, especially in terms of its ability to handle less frequent age groups.

C. Facial Region Extraction with MediaPipe FaceMesh

To enhance the utility of the dataset for age estimation, facial regions were extracted using MediaPipe's FaceMesh. This step involved segmenting each face into predefined regions, enabling the model to focus on localized features such as wrinkles, skin smoothness, and facial structure.

a) Technical Rationale for Using MediaPipe: MediaPipe FaceMesh was selected over alternatives like dlib due to its advantages in dense landmark detection, robustness to variability, and real-time processing capabilities. Compared to dlib's 68 landmarks, MediaPipe provides 468 landmarks, enabling more precise region extraction and facilitating the detection of additional regions, such as the chin or the forehead, which would be more challenging to obtain using dlib. This segmentation improvement is further illustrated in Figure 5. Additionally, MediaPipe demonstrates superior performance in handling the challenging conditions that surrounds this problem, such as pose and lighting variations.

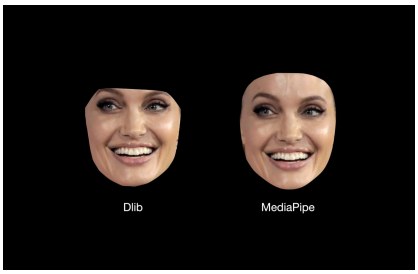

Fig. 5: Facial crop using dlib and MediaPipe

b) Definition of Landmarks: Each facial region was defined using specific subsets of MediaPipe landmarks:

- **Chin:** Key points defining the jawline (e.g., indices `[57, 136, 150]`).
- **Forehead:** Upper facial area landmarks (e.g., indices `[54, 68, 9]`).
- **Eyes:** Separate sets for left and right eyes.
- **Nose:** Points covering the bridge and nostrils.
- **Cheeks:** Points on both sides of the face defining the cheekbones.

These regions were selected based on their relevance to age-related features, ensuring that the extracted areas contained critical information for effective model training. One of the objectives of this approach is to train CNNs using multiple inputs, including the primary facial image and relevant facial regions where aging factors are more pronounced. This strategy aims to enhance the model's performance by leveraging the detailed information in these areas.

c) Implementation Pipeline: The following pipeline was implemented to process and extract facial regions:

1) **Face Detection:** Images were read and converted to RGB format for compatibility with MediaPipe.
2) **Landmark Detection:** MediaPipe FaceMesh detected 468 landmarks for each face. Images without detected faces were skipped

3) **Region Extraction:** Using the detected landmarks, bounding boxes for each predefined region were calculated, and the regions were cropped.
4) **Validation:** Images were saved only if all regions were successfully cropped and contained valid data.

d) Code Example: The Python implementation for processing the dataset is shown below:

```
def process_images(dataset_folder="dataset_rangos", start_folder="age_25_29"):
    face_mesh = mpFM(static_image_mode=True, max_num_faces=1, refine_landmarks=True)
    for age_folder in os.listdir(dataset_folder):
        ...
        if results.multi_face_landmarks:
            landmarks = [(int(l.x * image.shape[1]), int(l.y * image.shape[0]))
                         for l in results.multi_face_landmarks[0].landmark]
            regions_dict = create_face_regions(image, landmarks, landmarks_mediapipe_dict)
            save_regions(dataset_folder, age_folder, regions_dict, img_name)
```

The dataset size was significantly reduced as images without detectable faces were excluded. Consequently, two distinct datasets were created: one comprising the original unfiltered images and another consisting of facial sections, along with the full face images detected using MediaPipe. The second dataset is expected to have higher quality, as the included images were successfully processed by MediaPipe, ensuring that potential noise or distortions were not severe enough to hinder face detection by the model.

e) Output Structure: The output of this step consisted of directories organized by age ranges, with subdirectories for each facial region. This structure facilitated subsequent steps in the pipeline.

- Example directory structure:

```
dataset_rangos/
 age_25_29/
    original/
    chin/
    forehead/
    eyes/
    cheeks/
```

During the preprocessing stage, several challenges were encountered. One major issue was the inability to detect faces in some images due to occlusions or inadequate lighting conditions. To address this, images that failed to meet the detection criteria were excluded from the dataset to preserve its overall quality. Additionally, it was crucial to ensure that all extracted facial regions were valid and contained meaningful data. For this purpose, validation checks were implemented to confirm that each region met predefined quality standards before being saved. These measures ensured that the dataset maintained its integrity and was suitable for training robust models.

f) Impact of Preprocessing: This preprocessing step enhanced the dataset by segmenting facial features critical for age estimation. The structured and high-quality output enabled the subsequent model to learn localized age-related patterns more effectively. However, as shown in Figure 6, this step reduced the dataset size by approximately 50%. This significant reduction was caused by MediaPipe's inability to detect faces in certain images due to issues like poor lighting, occlusions, or low resolution. While the filtered dataset ensures higher quality, this drastic reduction exacerbates the imbalance in underrepresented age groups, posing additional challenges for model training.

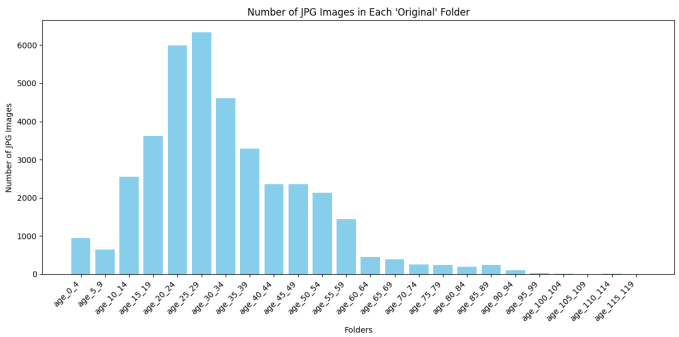

Fig. 6: Dataset distribution after applying MediaPipe segmentation, organized in 5-year ranges.

D. Multi-Input and Multichannel Models with Facial Parts

Two approaches were implemented to leverage additional facial features: multi-input and multichannel models.

a) Multi-Input Approach: Each facial part (e.g., chin, eyes, forehead) was processed independently using separate feature extractors, such as ResNet18. The extracted features were concatenated and passed through a fully connected layer for classification. This approach allowed each facial region to contribute unique, localized information to the model's decision-making process.

b) Multichannel Approach: In this approach, the original image and the extracted facial parts were combined into a single tensor with multiple channels. The combined tensor was then passed through a pre-trained MobileNetV2 architecture, treating the extracted parts as additional color channels. This method enabled the model to learn features holistically by leveraging both the full face and localized regions in a unified representation.

Both approaches presented certain challenges when leveraging transfer learning. The multi-input approach required processing multiple inputs, which increased computational and memory demands. Meanwhile, the multichannel approach faced difficulties due to the unconventional input tensor shape, which included $3 \times (n + 1)$ channels, where n represents the number of facial features and 1 accounts for the original face. These challenges, along with additional technical considerations are discussed in detail in the experiments section.

IV. Experiments

This section describes the experimental setup, including hardware and software configurations, as well as the constraints encountered during the project.

A. System Requirements and Limitations

The experiments were conducted on a system with the following specifications:

- **Operating System:** Windows 10 and Windows 11
- **GPU:** NVIDIA GeForce RTX 3060 (6GB VRAM)
- **RAM:** 16 GB
- **CUDA Version:** 11.8
- **cuDNN Version:** 11.8
- **Python Version:** 3.9 and 3.10
- **PyTorch Version:** 2.5.1
- **TensorFlow Version:** 2.13.0 and 2.15.0

B. Hardware and Software Constraints

One of the main challenges encountered during the project was the incompatibility between TensorFlow and the GPU setup. TensorFlow requires CUDA 11.8 and a compatible version of cuDNN, which is only fully supported on Windows 10 for GPU acceleration. Due to this limitation, it was not possible to configure TensorFlow to utilize the GPU, restricting its use to CPU-based computations. Consequently, all deep learning experiments done in the last part of this project were conducted using PyTorch, which provided full support for the existing CUDA version.

C. Initial Approach: Multichannel Networks

The initial approach aimed to utilize facial regions as additional input channels, allowing the model to extract more specific features related to age estimation. A multi-input convolutional neural network (CNN) was designed, where the input consisted of the original grayscale image concatenated with eleven additional channels corresponding to different facial regions (chin, mouth, eyebrows, etc.). This resulted in an input shape of $(224, 224, 12)$, which posed challenges in leveraging pre-trained models such as MobileNetV2 and ResNet50, as these architectures are optimized for three-dimensional tensors as inputs.

1) Memory Constraints and Tensor Overflows: Training a model with twelve input channels significantly increased memory usage, causing tensor overflows during the training process. The system, equipped with 16GB of RAM, struggled to handle large batch sizes, requiring a drastic reduction in batch size and dataset size to prevent crashes. Additionally, the high number of input channels prevented the direct application of transfer learning techniques, which are crucial for achieving good performance with limited datasets.

2) Mitigation Strategies: To address memory issues, several strategies were tested:

- **Reduced dataset size:** Instead of using the full dataset, only a subset (e.g., 10% and 50%) was used to alleviate memory consumption.
- **Conversion to grayscale:** As said before, instead of having 36 channels (3 color channels per image), this approach reduced the input to 12 channels (1 per image in grayscale per image), making the model computationally feasible.
- **Lower batch sizes:** The batch size was decreased from 32 to 16 and even to 8 in some experiments, to further reduce memory usage.

3) Training Experiments with Different Dataset Sizes: For this section of the experiments, the MobileNetV2 model was used, initialized with ImageNet weights. Three additional layers were added to adapt the neural network to this problem.

- A **Flatten** layer to convert convolutional features into a vector.
- A **Dense** layer with 128 neurons and ReLU activation.
- A **Dropout** layer (0.5 probability) to mitigate overfitting.
- A final **softmax** layer to classify the images into age groups.

The model was trained using **Adam** optimization with a learning rate of 0.0001 and sparse categorical cross-entropy as the loss function.

This series of experiments was conducted while varying the dataset size.

Figure 7, 8 and 9 presents the accuracy and loss curves for different dataset sizes.

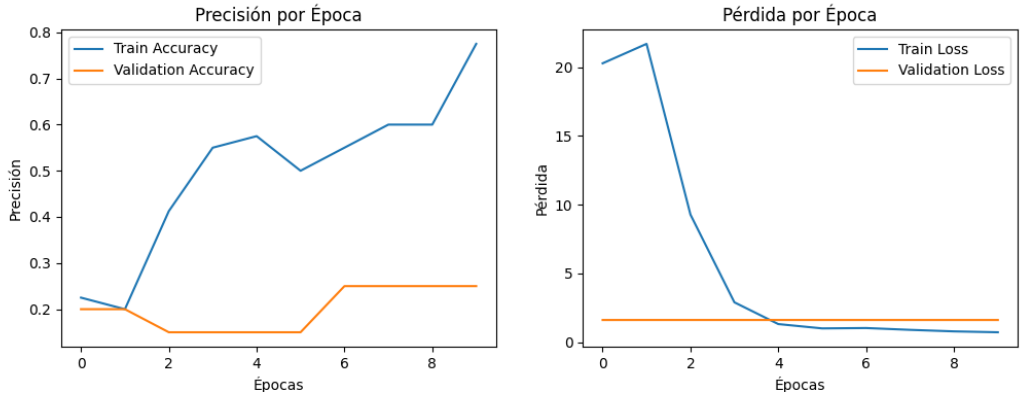

Fig. 7: Training performance with 100% of the dataset.

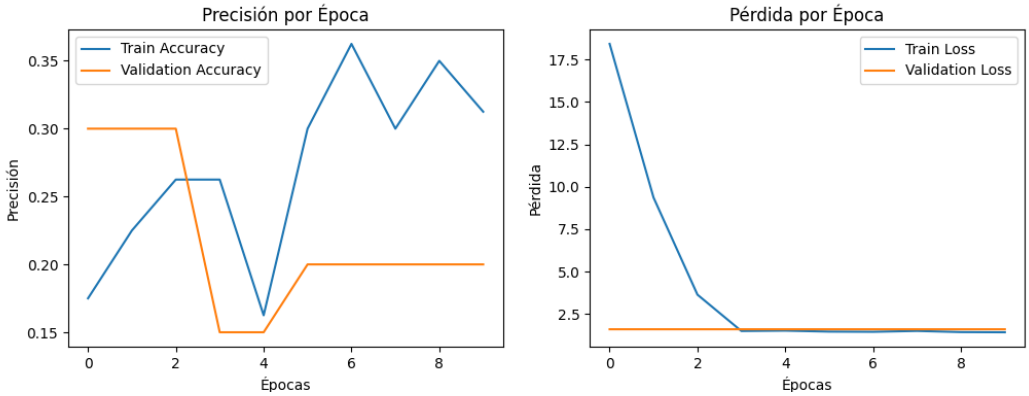

Fig. 8: Training performance with 50% of the dataset.

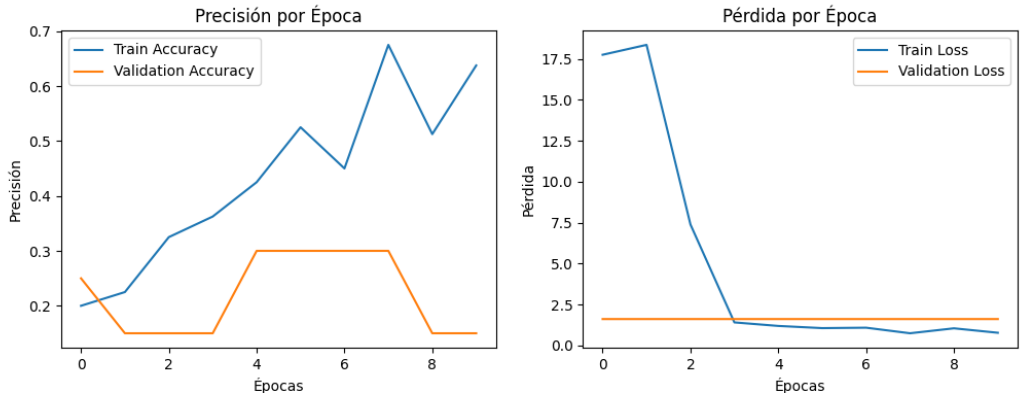

Fig. 9: Training performance with 10% of the dataset.

4) Analysis of Training Performance: The results reveal important insights into the impact of dataset size on training performance:

a) Accuracy per Epoch:

- **10% of the dataset:** The training accuracy increased rapidly, but the validation accuracy remained nearly constant and low (around 0.2 - 0.3). This indicates severe overfitting, where the model memorized the training data but failed to generalize.
- **50% of the dataset:** Training accuracy improved significantly. However, the validation accuracy was inconsistent, with sudden drops in certain epochs. This suggests that although the model had more data to learn from, it still struggled with generalization.
- **100% of the dataset:** The training accuracy continued to improve, but more gradually. The validation accuracy showed slight improvements but remained low overall. This suggests that increasing the dataset size alone was not sufficient to overcome overfitting.

b) Loss per Epoch: Across all training and validation loss plots, a similar pattern can be observed. The training loss decreases rapidly to very low values, while the validation loss remains nearly constant. This strongly suggests overfitting, as the model fails to generalize effectively beyond the training data.

5) Next Steps: Improving Multichannel Models: Despite the challenges encountered, multichannel models remain a promising approach for age estimation. However, the severe memory constraints present significant limitations. While there is still much to explore in this field, testing in this project was restricted due to both memory limitations and the extensive time required to build the data structures and train the models. Even with only 10 epochs and a limited number of experiments, the process proved to be too time-consuming.

Future research could explore the following alternatives to improve training efficiency:

- **Channel Selection and Optimization:** Instead of using grayscale images for all face regions, experiments could be conducted with RGB channels for specific regions or different pre-processing techniques to identify the most relevant features. Also, different color channels could be given as an input to the model.
- **Dimensionality Reduction:** Reducing the number of face regions or using principal component analysis (PCA) to compress information before feeding it to the model.
- **More Powerful Hardware:** The memory constraints encountered suggest that training on a GPU with higher memory capacity and more RAM memory could enable deeper architectures and larger batch sizes, leading to potentially better results.

6) Final Decision: Switching to Single-Input Face Images: Due to the high computational cost and limited improvements obtained from multichannel architectures, further testing was conducted using only the face as input. This decision was motivated by the need for a more feasible and scalable approach, allowing the direct application of pre-trained CNNs.

By switching to a single-input models, it was possible to train with larger datasets while leveraging transfer learning.

D. Transfer Learning with RGB Images

The goal of this group of experiments was to use pre-trained models, specifically MobileNetV2, Resnet50 and EfficientNet, which are optimized for three-channel RGB images. This approach aimed to improve computational efficiency while maintaining meaningful feature extraction.

1) Dataset Preparation and Label Mapping: To ensure a structured dataset, a class mapping was implemented. The dataset was split into three subsets:

- **Training set:** 64% of the total dataset (80% of the non-test data), used for optimizing model weights.
- **Validation set:** 16% of the total dataset (20% of the non-test data), used to monitor overfitting.
- **Test set:** 20% of the total dataset, used for final model evaluation.

This structured split ensured a balanced representation across different age groups and helped mitigate data leakage.

2) Model Architecture and Training Configuration: The model architecture consisted of MobileNetV2 with ImageNet pre-trained weights. The lower layers were initially frozen to retain generic feature extraction capabilities while allowing fine-tuning of the higher layers. The classification head included:

- A **Flatten** layer to convert convolutional features into a vector.
- A **Dense** layer with 128 neurons and ReLU activation.
- A **Dropout** layer (0.5 probability) to mitigate overfitting.
- A final **softmax** layer to classify the images into age groups.

The model was trained using **Adam** optimization with a learning rate of 0.0001 and sparse categorical cross-entropy as the loss function.

3) Experimental Results: Figure 10 presents the accuracy and loss curves during training.

a) Accuracy Analysis:

- The training accuracy shows a steady increase, reaching approximately **0.37 (37%)** by the end of training.
- Validation accuracy follows a similar pattern, peaking around **0.32 (32%)**, but exhibits some fluctuations beyond epoch 6.
- Around epoch 8, validation accuracy starts to stabilize, suggesting that the model's ability to generalize is limited.

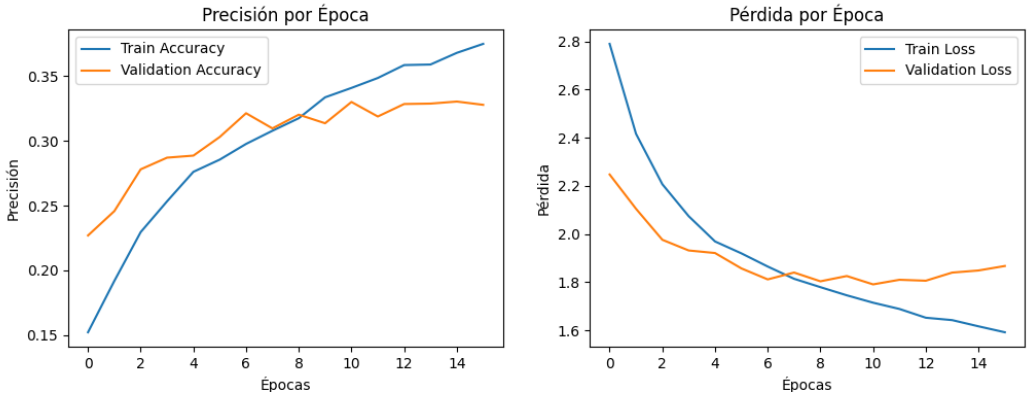

Fig. 10: Training performance using RGB images and transfer learning with MobileNetV2.

b) Loss Analysis:

- The training loss decreases consistently, indicating that the model is effectively minimizing errors on the training set.
- Validation loss follows a downward trend until around epoch 6, after which it fluctuates slightly and begins to stabilize.
- A slight increase in validation loss beyond epoch 10 suggests the onset of overfitting, as the model begins to specialize in training data rather than generalizing to unseen samples.

c) Observations and Limitations:

- The model tries to learns patterns related to age estimation, as evidenced by increasing accuracy and decreasing loss.
- However, the gap between training and validation accuracy increases after epoch 10, highlighting overfitting tendencies.
- Due to these results, no further training was conducted using this approach, as it was evident that increasing the number of epochs would only lead to overfitting. Since the models did not show promising results, they were not evaluated on the test data, as the validation data was sufficient to determine that this approach was ineffective.

E. Fine-Tuning with ResNet50 and Data Augmentation

To further improve performance, this experiment leveraged transfer learning with **ResNet50**, incorporating data augmentation and class balancing techniques. The goal was to enhance model generalization and mitigate the impact of dataset imbalance.

a) Model Configuration: The model was trained with the following configuration:

- **Architecture:** Pretrained **ResNet50** from ImageNet.
- **Batch Size:** 32.
- **Learning Rate:** 0.001.
- **Head Structure:** The same fully connected layers as in previous CNN models:
  - **Flatten layer**.
  - **Dense layer** with 256 neurons and ReLU activation.
  - **Output layer** with as many neurons as age categories, using softmax activation.

b) Data Augmentation Strategy: To improve model generalization and reduce overfitting, data augmentation was applied to the training images. The augmentation transformations included:

- **Random rotations** within a range of $\pm 15°$.
- **Horizontal flips** to increase variability.
- **Brightness adjustments** within a range of $[-30, 30]$ pixel intensity.
- **Histogram equalization** to normalize contrast across images.
- **Denoising filtering** using Non-Local Means Denoising to reduce noise while preserving facial details.

These augmentations were applied consistently to all images, without specifically targeting minority classes. While this helps generalization, future improvements could explore adaptive augmentation that prioritizes underrepresented classes.

c) Class Balancing: Since the dataset suffered from class imbalance, class weights were computed and applied during training. The weights were calculated using Scikit-Learn's compute_class_weight function, assigning higher weights to classes with fewer samples. These weights were then used in the categorical cross-entropy loss function, ensuring that misclassifications in underrepresented classes had a greater impact on model optimization.

d) Results: The accuracy and loss curves obtained for this model are shown in Figure 11.

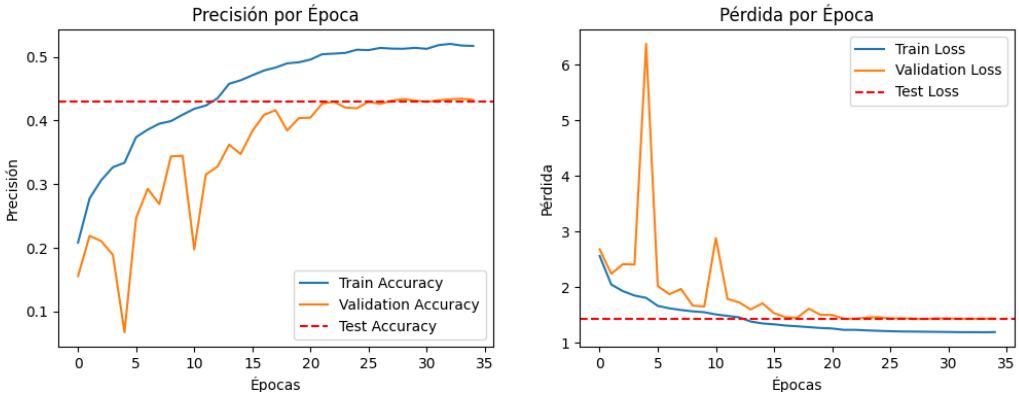

Fig. 11: Training, validation, and test performance with ResNet50 fine-tuning.

e) Accuracy Analysis:

- **Training accuracy** steadily increases, reaching approximately **0.52 (52%)**, demonstrating the model's capacity to learn.
- **Validation accuracy** follows an irregular pattern, initially fluctuating significantly before stabilizing around **0.42 (42%)**.
- The **test accuracy**, represented by the dashed red line, aligns closely with the validation accuracy, indicating that the model maintains its performance on unseen data.
- The initial fluctuations in validation accuracy suggest instability in early epochs, likely due to the introduction of fine-tuning in deeper layers.

f) Loss Analysis:

- **Training loss** decreases consistently, suggesting effective learning.
- **Validation loss** initially exhibits high variance, including significant spikes, before stabilizing around epoch 20.
- The **test loss**, indicated by the red dashed line, is comparable to the stabilized validation loss, reinforcing the generalization ability of the model.
- The spikes in validation loss during early epochs may indicate learning rate instability or the impact of fine-tuning deeper layers without sufficient pre-training adaptation.

g) Observations and Model Behavior:

- The class balancing strategy using data augmentation appears to improve overall performance by providing additional synthetic samples for underrepresented classes.
- Despite fluctuations in early epochs, the model stabilizes, suggesting that gradual layer unfreezing and learning rate scheduling helped fine-tuning.
- The gap between training and validation accuracy is moderate, indicating that the model generalizes reasonably well, though further adjustments could enhance performance.

h) Results Summary and Next Steps: The experiments conducted so far have highlighted both the challenges and improvements associated with different training strategies. The transition from a multi-channel approach to a transfer learning-based model significantly reduced memory consumption and enabled the use of pre-trained networks such as MobileNetV2 and ResNet50, with the latter yielding the best results.

i) Key Observations:

- **Improved Training Stability**: The use of transfer learning resulted in more stable training curves compared to previous multi-channel models, which suffered from memory issues and tensor overflows.
- **Class Imbalance Handling**: Data augmentation techniques successfully improved class balance, mitigating the impact of underrepresented age groups.
- **Validation Fluctuations**: The fine-tuned ResNet50 model exhibited fluctuations in validation accuracy during early epochs, stabilizing over time.
- **Test Performance Alignment**: The test accuracy aligned closely with validation accuracy, suggesting that the model's performance generalizes reasonably well.
- **Accuracy values**: While the model achieved stability, its accuracy could still be improved. The balancing techniques provided a solid foundation, but they also introduced new challenges in the architecture that need to be addressed to further enhance performance.

j) Next Experimental Direction - Generalizing Age Labels: Despite these improvements, the results indicate that age classification remains a complex task, with models struggling to achieve high accuracy. One potential limiting factor is the granularity of the age labels. The current classification method assigns labels in 5-year intervals (e.g., 0-4, 5-9, 10-14, etc.), which may introduce noise due to intra-group variations that blur the boundaries between consecutive classes.

To address this, the next phase of experimentation will explore a more generalized labeling strategy, where age groups are defined in **10-year intervals** (e.g., 0-9, 10-19, etc.). This modification aims to:

- **Reduce inter-class confusion**, as larger groups may help smooth label noise.
- **Improve model generalization**, by providing a broader representation of age characteristics per class.
- **Simplify the classification task**, making it easier for the model to distinguish between groups.

This adjusted labeling approach will be implemented in subsequent experiments to evaluate whether broader class definitions contribute to improved accuracy and robustness.

k) Discussion and Next Steps: This experiment demonstrated that using RGB images and transfer learning significantly improved training stability compared to the multi-channel approach with the current limitations. The structured dataset split provided a more reliable evaluation framework. However, the model's accuracy remained relatively low, suggesting that further improvements were needed.

Potential next steps include:

- Fine-tuning deeper layers of MobileNetV2 for enhanced feature extraction.
- Experimenting with different pre-trained architectures (e.g., ResNet50, EfficientNet).
- Adjusting class groupings to determine the optimal level of age granularity.

Despite its limitations, this experiment laid the groundwork for refining the age estimation pipeline and optimizing the balance between model complexity and generalization.

F. Dataset Preparation: Addressing Class Imbalance

Following the transition to age classification in 10-year intervals, the dataset was reorganized into three different versions to evaluate the impact of class distribution on model performance:

- **Original Dataset:** Contains the unaltered dataset with natural class distributions.
- **Balanced Dataset:** Applies data augmentation to underrepresented classes, ensuring that each class has the same number of images.
- **Face-Detected Dataset:** Uses MediaPipe face detection to extract clear, cropped facial images, removing background noise and improving facial focus.

a) Class Imbalance and Dataset Selection: One of the primary challenges observed in the original dataset was the severe class imbalance. Certain age groups contained fewer than 500 images, while others had more than 10,000 images. This discrepancy introduced significant bias during training, as models would naturally favor majority classes.

To address this, the balanced dataset was created by artificially augmenting underrepresented classes until all classes had the same number of samples. However, this approach presented some drawbacks:

- The augmented images introduced artificial patterns that could mislead the model.
- The synthetic balance did not fully reflect the real-world age distribution, potentially reducing generalization.
- Some classes underwent extensive data augmentation, while others received none, resulting in a non-homogeneous dataset.

Given these drawbacks, the balanced dataset was quickly discarded from further experiments. Instead, the focus was placed on the original dataset and the face-detected dataset, as they provided a more realistic representation of age distribution while maintaining practical training times.

The next set of experiments evaluates how these two datasets impact classification performance using the CNNs EfficientNet-B4 and EfficientNet-B5.

G. Experimental Results with EfficientNet-B4

1) Training Process: The model was trained using EfficientNet-B4 as the backbone, leveraging the FastAI framework. The dataset was structured into 10-year age range bins, with predefined splits for training, validation, and testing. The training process followed a **one-cycle policy**, gradually increasing and then decreasing the learning rate to enhance convergence.

During the training phase, the accuracy steadily improved over 60 epochs. The key accuracy milestones were:

- **Epoch 0**: Initial accuracy of **41.6%**.
- **Epoch 10**: Accuracy reached **59.4%**, showing consistent learning progress.
- **Epoch 25**: Accuracy reached **63.6%**, demonstrating stable learning.
- **Epoch 45**: Accuracy continued improving, reaching **65.7%**.
- **Epoch 57**: The best accuracy was recorded at **65.8%**, after which no significant improvements were observed.

The **learning rate finder** identified an optimal learning rate, which was used throughout training. The model automatically saved the best weights based on validation accuracy.

2) Loss and Accuracy Analysis:

- **Accuracy Trends**: The validation accuracy closely followed the training accuracy, suggesting minimal overfitting. The final validation accuracy stabilized at **65.8%**.
- **Loss Trends**: The training loss steadily decreased, indicating convergence. The validation loss also followed a downward trend, stabilizing at a reasonable level.
- **MAE (Mean Absolute Error)**: The final MAE on the test set was **0.4723**, suggesting that the model's age predictions, on average, were approximately **4.72 years off**.

3) Test Set Performance: After training, the best model was loaded and evaluated on the test set. The **final test accuracy was 65.87%**, with a test loss of **0.8469**.

- **Precision and Recall**: The model achieved relatively high precision and recall for well-represented age groups (e.g., age_20_29 and age_10_19).
- **Older Age Groups**: The performance decreased significantly for classes with fewer samples, such as age_70_79 and age_90_99, where recall values were notably lower.
- **Misclassifications**: Some misclassifications occurred between adjacent age groups, indicating that the model still struggles with fine age distinctions.

4) Confusion Matrix Analysis: Figure 12 presents the confusion matrix for the test set. Several key insights emerge:

- **Majority of correct predictions**: The highest concentration of correct predictions lies along the diagonal, meaning the model generally predicts the correct age category.
- age_20_29 and age_10_19 were the best-predicted classes: These classes had the largest sample sizes, reinforcing the impact of dataset balance.
- **Poor Performance in Underrepresented Classes**: Age groups such as age_100_109 and age_110_119 contained very few examples, making it difficult for the model to generalize. Consequently, predictions for these classes were unreliable.
- **Significant Performance Drop in Older Age Groups**: Classes above age_60 exhibited noticeably lower classification accuracy. This suggests that either facial features become harder to distinguish at older ages, or the severe data imbalance hindered the model's ability to generalize.
- **Adjacent Age Group Confusion**: Many samples were misclassified into neighboring age groups (e.g., age_40_49 was often confused with age_30_39 and age_50_59).

This behavior aligns with expectations, as distinguishing between adjacent age groups is inherently challenging due to natural aging variations.

H. Improvements in Training Strategy

After the initial experimentation with EfficientNetB4 and a basic training approach, several limitations were identified that affected the model's performance and generalization. The primary issues observed included class imbalance, training instability, and lack of control over the learning process. To address these problems, a series of modifications were introduced in the next iteration of the training pipeline.

1) Optimization and Regularization Enhancements: One of the major adjustments was replacing the default optimizer with **AdamW**. Unlike standard Adam, AdamW decouples weight decay from the learning rate, leading to better generalization and reducing overfitting. Additionally, a specific weight decay factor (`wd=1e-2`) was incorporated to further regularize the model. This change was crucial in preventing the model from excessively memorizing the training data while improving stability during gradient updates.

2) Improved Training Management: In the initial approach, the training process lacked a mechanism to halt training when performance stopped improving. To address this, an **Early Stopping Callback** was introduced with a patience of 10 epochs. This prevents unnecessary training cycles and reduces overfitting risks by stopping training once accuracy ceases to improve. Additionally, a **learning rate finder** was used to automatically select an optimal learning rate, enhancing convergence speed.

3) Addressing Class Imbalance: One of the most critical issues in age estimation is the imbalance across age groups, with some classes containing thousands of images while others have only a few samples. To mitigate this, **class weighting** was introduced in the loss function. The class weights were calculated as the inverse of the number of samples per class and applied to the `CrossEntropyLossFlat` function. This modification aimed to reduce bias toward overrepresented classes and improve performance in underrepresented age groups.

4) Expanded Model Evaluation: While the first approach mainly relied on accuracy and the mean absolute error (MAE), the improved version included additional metrics:

- **Balanced Accuracy Score:** Accounts for class imbalance by computing accuracy per class and averaging the results.
- **Cohen's Kappa Score:** Measures agreement between predicted and actual labels, making it particularly useful for ordinal classification tasks like age estimation.

These additional metrics provide a more comprehensive evaluation of the model's real performance, ensuring that accuracy alone does not misrepresent the quality of predictions.

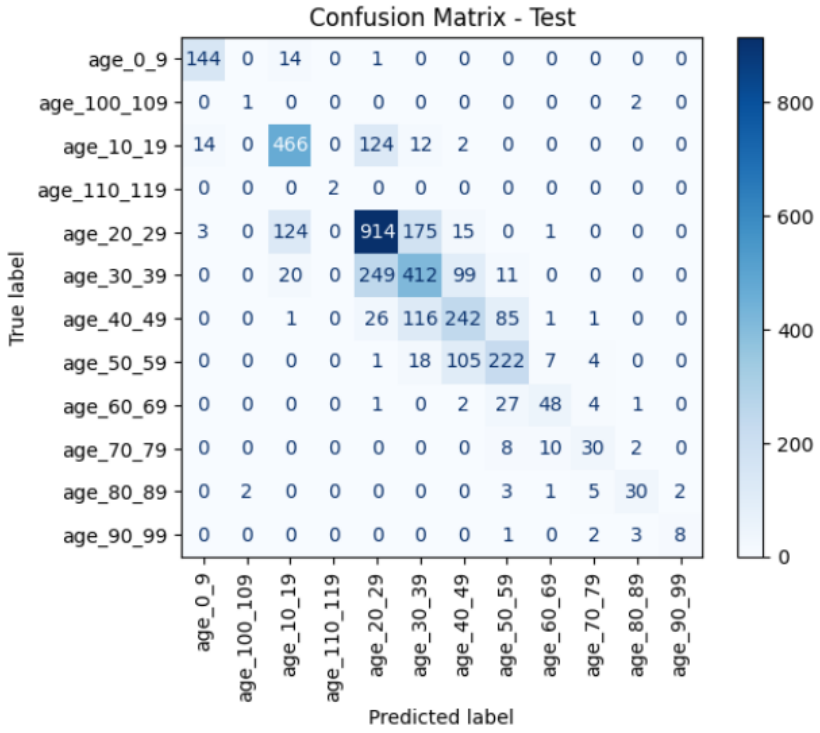

Fig. 12: Confussion matrix of efficientnetb4 using FastAI.

Clasificación detallada (test):

	precision	recall	f1-score	support
age_0_9	0.89	0.91	0.90	159
age_100_109	0.33	0.33	0.33	3
age_10_19	0.75	0.75	0.75	618
age_110_119	1.00	1.00	1.00	2
age_20_29	0.69	0.74	0.72	1232
age_30_39	0.56	0.52	0.54	791
age_40_49	0.52	0.51	0.52	472
age_50_59	0.62	0.62	0.62	357
age_60_69	0.71	0.58	0.64	83
age_70_79	0.65	0.60	0.62	50
age_80_89	0.79	0.70	0.74	43
age_90_99	0.80	0.57	0.67	14
accuracy			0.66	3824
macro avg	0.69	0.65	0.67	3824
weighted avg	0.66	0.66	0.66	3824

Fig. 13: Table of metrics of each label for EfficientNet-B4 using FastAI.

5) Memory Management and GPU Optimization: To enhance computational efficiency, the second iteration incorporated `torch.cuda.empty_cache()` before training. This explicitly clears unused GPU memory, reducing potential bottlenecks and preventing crashes in environments with limited GPU resources.

6) Evaluation and Comparison of Results: The improved training strategy led to a series of modifications, including weighted loss functions, an optimized optimizer, and better control over learning stability. The following section presents a detailed analysis of the obtained results and a comparison with the previous implementation.

7) Training Progress: The training process showed a steady improvement in validation accuracy, with notable milestones:

- The model started with an initial accuracy of **23.3%** and gradually improved over the epochs.
- A significant increase was observed up to epoch 10, where accuracy reached **47.1%**, indicating effective learning during the early stages.

- After epoch 15, the accuracy improvement slowed down, showing fluctuations but still increasing steadily.
- The best accuracy was reached at **epoch 72**, with a validation accuracy of **61.2%**, suggesting that the model continued learning but with diminishing returns.
- During the last 20 epochs, the model's validation accuracy only increased from 59.1% to 60.7%, suggesting that the model had stabilized.

8) Test Performance Analysis: The model achieved a test accuracy of **60.43%**, showing a slight decrease compared to the previous approach but with improvements in class balance metrics, as shown in Figure 15. The Mean Absolute Error (MAE) also increased to **0.5845**, indicating that, on average, predictions were within approximately half a class range of the correct category.

- **Test Loss:** The final loss on the test set was **0.8469**, suggesting that the model generalizes relatively well to unseen data.
- **Macro Average Scores:** The model achieved a macro-averaged precision of **68%** and recall of **67%**, confirming balanced performance across most classes.
- **Weighted Average Scores:** With an overall weighted accuracy of **61%**, the model maintained stable performance across both frequent and less common age groups.
- **Cohen's Kappa:** The model achieved a Cohen's Kappa score of **0.5137**, indicating moderate agreement between predictions and actual labels.

9) Confusion Matrix Analysis: The confusion matrix (Figure 14) reveals critical insights about the model's predictions:

- **Strong performance in well-represented classes:** Age groups such as `age_10_19`, `age_20_29`, and `age_30_39` achieved high precision and recall, benefiting from a large number of training samples.
- **Difficulties in underrepresented age groups:** Similar to the last experiment, classes such as `age_100_109` and `age_110_119` were poorly predicted due to their low representation in the dataset.
- **Expected confusion in neighboring classes:** As seen in previous experiments, the model struggles to distinguish adjacent age categories, especially between `age_30_39` and `age_40_49`, which is expected given the visual similarities between consecutive age groups.
- **General improvements in overall classification:** Compared to the previous model, which exhibited more instability in classification results, this iteration significantly reduced misclassifications, particularly in middle-aged groups.

10) Comparison with Previous Model: Several key differences emerged when comparing this approach to the initial training setup:

- **Better training stability:** The addition of weight decay and AdamW prevented overfitting while maintaining a more stable learning curve.
- **Balanced Accuracy:** The model achieved a **balanced accuracy of 66.75%**, which is an improvement in class balance despite a slight reduction in overall accuracy.
- **Higher test accuracy in underrepresented classes:** Despite a slight drop in overall accuracy, some underrepresented classes improved due to the use of class-weighted loss functions.
- **Cohen's Kappa improvement:** The Cohen's Kappa score of **0.5137** confirms that predictions were more consistent across classes.

11) Summary of Results: Compared to the previous experiment, where validation accuracy reached **65.87%**, the new setup achieved a lower final accuracy of **60.43%**. Additionally, the Mean Absolute Error (MAE) increased from **0.4723** to **0.5845**, indicating that predictions were less precise.

While the introduction of **AdamW** and **class-weighted loss functions** aimed to improve training stability, the results suggest that the learning process stagnated rather than stabilizing. Over the last 20 epochs, validation accuracy only improved from **59.1%** to **60.7%**, showing diminishing returns in learning.

Furthermore, **balanced accuracy** decreased from **68.86%** to **66.75%**, and **Cohen's Kappa** also dropped, indicating a decline in classification reliability across classes. These findings suggest that, although weight balancing and AdamW may have contributed to mitigating extreme fluctuations, they did not necessarily lead to a more controlled or effective learning process. Instead, performance slightly worsened compared to the previous approach.

I. Last experiment - EfficientNet-B5

After evaluating the performance of the model trained with **EfficientNet-B4**, several improvements were introduced to enhance accuracy, generalization, and robustness against class imbalance. These changes were implemented in the new version of the experiment, which utilizes **EfficientNet-B5**. The modifications are detailed below:

1) Architectural Change: EfficientNet-B4 to EfficientNet-B5: The initial experiments employed **EfficientNet-B4** as the backbone architecture. In this new approach, **EfficientNet-B5** was chosen due to its higher capacity and increased number of parameters, allowing for better feature extraction and improved classification performance.

Rationale: EfficientNet-B5 extends the capabilities of its predecessor by leveraging a deeper network, making it more suitable for complex classification tasks such as age estimation.

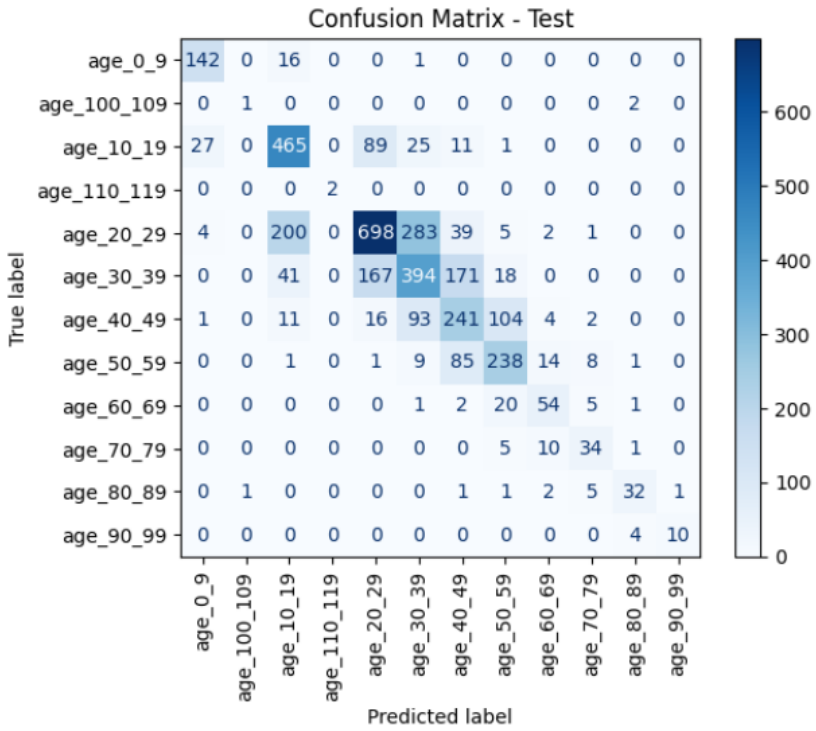

Fig. 14: Confusion matrix of the improved model on the test set.

Clasificación detallada (test):

	precision	recall	f1-score	support
age_0_9	0.82	0.89	0.85	159
age_100_109	0.50	0.33	0.40	3
age_10_19	0.63	0.75	0.69	618
age_110_119	1.00	1.00	1.00	2
age_20_29	0.72	0.57	0.63	1232
age_30_39	0.49	0.50	0.49	791
age_40_49	0.44	0.51	0.47	472
age_50_59	0.61	0.67	0.64	357
age_60_69	0.63	0.65	0.64	83
age_70_79	0.62	0.68	0.65	50
age_80_89	0.78	0.74	0.76	43
age_90_99	0.91	0.71	0.80	14
accuracy			0.60	3824
macro avg	0.68	0.67	0.67	3824
weighted avg	0.61	0.60	0.61	3824

Fig. 15: Table of metrics of each label for EfficientNet-B4 using FastAI.

2) Class Weighting: Addressing Class Imbalance: One of the main challenges in age estimation is the **class imbalance**, where certain age groups contain significantly more samples than others.

- **Previous approach:** Used standard cross-entropy loss without any class weighting, leading to a bias toward overrepresented classes.
- **Current approach:** Introduces **weighted loss** using class frequencies. The model assigns higher importance to underrepresented classes by adjusting the loss function accordingly.

This technique improves performance on minority classes, reducing the tendency of the model to favor majority classes and enhancing overall classification fairness.

3) Optimization: AdamW: For this experiment, the optimization of AdamW was chosen again despite the last results. This is because it's a better optimazer for the current problem

- **AdamW settings:** A weight decay of 10^{-2} was applied, along with *decoupled weight decay* to separate weight decay

from the learning rate.

Benefit: AdamW reduces overfitting and improves convergence, making the training process more stable.

4) Enhanced Training Strategy: To further refine the training process, these mechanisms were implemented:

- **Early Stopping:** Introduced with a patience of 10 epochs to prevent overfitting.
- **SaveModelCallback:** Ensures that the best model, based on validation accuracy, is stored.

This prevents unnecessary training beyond the point of diminishing returns, ensuring an optimal balance between training duration and model performance.

5) Evaluation Metrics: The evaluation process was enhanced with these performance metrics, providing a comprehensive assessment of model performance beyond traditional accuracy scores:

- **Balanced Accuracy:** Measures accuracy across all classes equally, particularly useful for imbalanced datasets.
- **Cohen's Kappa Score:** Evaluates the agreement between predicted and true labels, adjusting for chance predictions.

6) Training Performance: The training process for EfficientNet-B5 showed steady improvement over the epochs, reaching its best validation accuracy at epoch 57 before early stopping was triggered. The following key observations can be made:

- **Initial learning phase:** The model exhibited a gradual increase in validation accuracy, starting from 0.233 at epoch 0 and surpassing 0.50 around epoch 15.
- **Stable progression:** After epoch 30, improvements in accuracy slowed, with a final validation accuracy of **0.617**.
- **Early stopping:** The model stopped improving at epoch 57, suggesting that further training would not yield significant benefits.

Compared to the previous experiment using EfficientNet-B4, this model showed a smoother convergence and improved generalization due to the applied modifications, including class balancing and weight decay.

7) Test Set Evaluation: The evaluation on the test set resulted in the following performance metrics:

- **Test Accuracy: 62.16%**, which represents a slight decline from the best experiment (**0.6587**).
- **Mean Absolute Error (MAE): 0.5570**, which is higher than first experiment of EfficientNet-B4 (**0.4723**), indicating that predictions deviated more from the correct class.
- **Balanced Accuracy: 0.6886**, a significant improvement compared to the previous run, showing better performance across all classes.
- **Cohen's Kappa: 0.5327**, which measures agreement between predictions and ground truth, indicating moderate reliability.

8) Confusion Matrix Analysis: Figure 16 presents the confusion matrix for the EfficientNet-B5 model.

The confusion matrix highlights and the data shown in the Figure 17 have several key aspects:

- **High accuracy in younger age groups:** The *age_0_9* category maintains a high recall of **92%**, indicating strong classification performance for this age group.
- **Improvements in older age groups:** The model demonstrates improved recall for underrepresented age groups, such as *age_60_69* and *age_70_79*, where precision and recall are higher than in previous experiments.
- **Class misclassification trends:**
  - The *age_20_29* category remains the most frequently predicted class, absorbing misclassifications from nearby age groups. However, its performance is slightly worse compared to previous experiments.
  - Confusion between *age_30_39* and *age_40_49* persists, reflecting the difficulty in distinguishing between visually similar facial features.

This experiment was the last conducted in this project, covering multiple neural networks where transfer learning was applied, as well as different data distributions and input types.

V. Conclusions and Future Work

The experiments conducted in this project explored different approaches to age estimation using deep learning, each with its strengths and limitations. The main takeaways from these experiments are as follows:

A. Key Findings

- **Multichannel Architectures:** The initial approach using multichannel models proved impractical due to excessive memory consumption and incompatibility with pre-trained networks. Although theoretically promising, the computational cost made them infeasible for further experiments.
- **Transfer Learning with 3-Channel Input:** Implementing transfer learning with MobileNetV2 and ResNet50 significantly improved efficiency and model convergence. However, class imbalance remained a major challenge, leading to poor generalization for underrepresented age groups.
- **FastAI-Based Training:** Transitioning to FastAI allowed for more structured training, optimizing data loading and augmentation. The implementation of AdamW, class-weighted loss functions, and EfficientNet-B4 yielded the highest

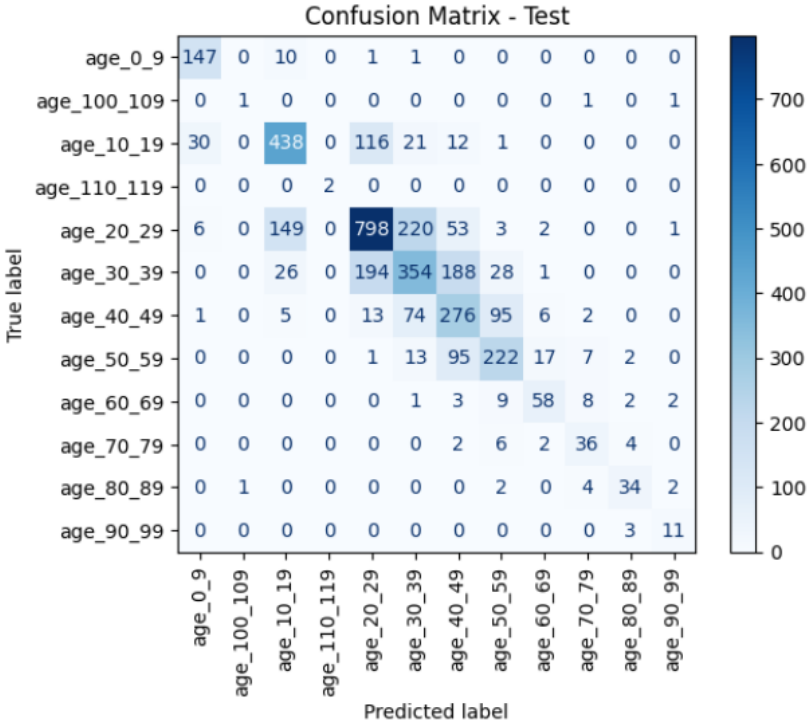

Fig. 16: Confusion Matrix for EfficientNet-B5 on Test Set

Clasificación detallada (test):

	precision	recall	f1-score	support
age_0_9	0.80	0.92	0.86	159
age_100_109	0.50	0.33	0.40	3
age_10_19	0.70	0.71	0.70	618
age_110_119	1.00	1.00	1.00	2
age_20_29	0.71	0.65	0.68	1232
age_30_39	0.52	0.45	0.48	791
age_40_49	0.44	0.58	0.50	472
age_50_59	0.61	0.62	0.61	357
age_60_69	0.67	0.70	0.69	83
age_70_79	0.62	0.72	0.67	50
age_80_89	0.76	0.79	0.77	43
age_90_99	0.65	0.79	0.71	14
accuracy			0.62	3824
macro avg	0.66	0.69	0.67	3824
weighted avg	0.63	0.62	0.62	3824

Fig. 17: Table of metrics of each label for EfficientNet-B5 using FastAI.

test accuracy (65.87%). Despite this, the model struggled with class imbalance and tended to misclassify adjacent age groups.

- **EfficientNet-B5 Comparison:** The shift to EfficientNet-B5 was intended to leverage a more complex model architecture. However, results showed a decrease in overall accuracy (from 65.87% to 62.16%) and an increase in MAE, suggesting that the additional complexity did not provide the expected benefits.

B. Comparison between EfficientNet-B4 and EfficientNet-B5

Table I summarizes the differences between the two models.
Key takeaways from this comparison:

Metric	EfficientNet-B4	EfficientNet-B5
Test Accuracy	0.6587	0.6216 (-3.7%)
MAE	0.4723	0.5570 (+8.4%)
Balanced Accuracy	0.65	0.6886 (+3.8%)
Cohen's Kappa	Not measured	0.5327

TABLE I: Comparison of EfficientNet-B4 and EfficientNet-B5 Models

- **Better class balance:** The EfficientNet-B5 model improved balanced accuracy by 3.8%, meaning it performed more evenly across classes.
- **Slight decline in test accuracy:** Despite better balance, the overall test accuracy dropped by 3.7%, likely due to reduced bias toward majority classes.
- **Higher MAE:** The increased MAE suggests that predictions were less precise, possibly due to class balancing techniques increasing prediction variance.

C. Future Work

Based on these findings, several aspects could be explored to further improve age estimation models:

- **Advanced Data Augmentation:** Implementing generative models (e.g., GANs) or changing the color channels of the data could help with the lack of data.
- **Alternative Model Architectures:** Testing other architectures such as Vision Transformers (ViTs) or hybrid models may provide better generalization.
- **Multi-Modal Learning:** Exploring additional input features (e.g., texture, depth maps, or facial landmarks) could improve performance beyond standard RGB images.
- **Post-Processing Techniques:** Applying ensemble methods or calibration techniques could reduce classification variance and enhance robustness.
- **Fine-Tuned Loss Functions:** Experimenting with focal loss or ordinal regression-based loss functions may mitigate misclassifications between adjacent age groups.

D. Final Remarks

The progression from multichannel architectures to FastAI-based training demonstrated the importance of balancing model complexity, training efficiency, and class distribution. While EfficientNet-B4 yielded the best results, future work should focus on refining data distribution, loss functions, and model selection to further enhance accuracy and robustness. These findings provide a strong foundation for continued research in deep learning-based age estimation.

APPENDIX A
LINKS TO THE EXPERIMENT FILES

The GitHub repository containing the code and implementations for this work can be accessed at: https://github.com/JaviertFM/AgeRecognition

The datasets used in the experiments and the models obtained are available at: https://drive.google.com/drive/folders/1_OFxvbr6ozinRFYRqjyU9iypyV9aohzF?usp=sharing

REFERENCES

[1] Salim Abdullah et al. "Age estimation using deep learning for security and surveillance applications". In: *International Journal of Advanced Computer Science and Applications* (2021).

[2] Alfonso Alcañiz. "Super resolución facial basado en Deep Learning". MA thesis. Universitat Politècnica de València, 2020. URL: https://riunet.upv.es/bitstream/handle/10251/175078/Alcaniz%20-%20Super%20resolucion%20facial%20basado%20en%20Deep%20Learning.pdf?sequence=1.

[3] J. Alcázar, A. García, and J. González. "Deep Facial Model for Face Mask Application". In: *Journal of Computer Vision* 45 (2020). Accessed: 2025-01-14, pp. 123–135.

[4] Author(s). *Confusion Matrix for Age Estimation*. Accessed: 2025-01-14. 2023. URL: https://www.researchgate.net/.

[5] Author(s). "Title of the article". In: *PMC* (2023). Accessed: 2025-01-14. URL: https://pmc.ncbi.nlm.nih.gov/.

[6] Author(s). *What are the disadvantages of accuracy?* Accessed: 2025-01-14. 2023. URL: https://datascience.stackexchange.com/questions/110124/what-are-the-disadvantages-of-accuracy.

[7] Paula Branco et al. "A survey of predictive modeling on imbalanced domains". In: *ACM Computing Surveys* (2016).

[8] A. Bulat and G. Tzimiropoulos. "Accurate facial landmark localization with convolutional neural networks". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 1022–1030.

[9] Qing Cao, Linlin Yin, and Xiaoou Tang. “Cross-age face recognition: A survey and evaluation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2010).

[10] X. Cao et al. “Face alignment at 3000 FPS via regressing local binary features”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.3 (2014), pp. 679–690.

[11] Bor-Chun Chen, Chu-Song Chen, and Yi-Ping Hsu. “Cumulative attribute-based representation for age inference and retrieval”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2013).

[12] Albert Clapés et al. “From apparent to real age: gender, age, ethnic, makeup, and expression bias analysis in real age estimation”. In: *proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2018, pp. 2373–2382.

[13] “Complexities in Age Estimation Models”. In: *PubMed* (). Accessed: YYYY-MM-DD. URL: https://pubmed.ncbi.nlm.nih.gov/39733693/.

[14] T. F. Cootes et al. “Active Shape Models—Their Training and Application”. In: *Computer Vision and Image Understanding* 61.1 (1995), pp. 38–59.

[15] Rui Ding et al. “Artificial intelligence in age-related disease diagnosis: A review”. In: *Nature Machine Intelligence* (2022).

[16] Guodong Guo and Yunfu Mu. “Human age estimation through multitask learning with binary labels”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2009).

[17] Guodong Guo and Yunfu Mu. “Image-based human age estimation by manifold learning”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* (2008).

[18] K. He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[19] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning”. In: *Nature* (2015).

[20] Y. LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* (1998).

[21] G. Levi and T. Hassner. “Age and gender classification using convolutional neural networks”. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2015).

[22] Gil Levi and Tal Hassner. “Age and gender classification using convolutional neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2015, pp. 34–42.

[23] “Limitations of Linear and Support Vector Models in Age Estimation”. In: (). Accessed: YYYY-MM-DD. URL: https://stats.stackexchange.com/questions/633091/support-vector-regression-vs-linear-regression.

[24] J. Liu et al. “LCA-GAN: Learning Collaborative Attention for Facial Expression Recognition”. In: *IEEE Transactions on Affective Computing* (2021).

[25] Ilya Loshchilov and Frank Hutter. “SGDR: Stochastic gradient descent with warm restarts”. In: *International Conference on Learning Representations (ICLR)* (2016).

[26] M. I. M. Navas, J. Gutiérrez, and other authors. *Evaluación y desarrollo de métodos para la estimación de edad de la población colombiana*. Diposit, Universidad de Barcelona. 2020. URL: https://diposit.ub.edu/dspace/bitstream/2445/149998/1/EVMS_TESIS.pdf.

[27] W. S. McCulloch and W. Pitts. “A logical calculus of the ideas immanent in nervous activity”. In: *Bulletin of Mathematical Biophysics* (1943).

[28] A. Mollahosseini, B. Hasani, and M. H. Mahoor. “AffectNet: A database for facial expression, valence, and arousal computing in the wild”. In: *IEEE Transactions on Affective Computing* (2017).

[29] S. J. Pan and Q. Yang. “A survey on transfer learning”. In: *IEEE Transactions on Knowledge and Data Engineering* (2009).

[30] O. M. Parkhi, A. Vedaldi, and A. Zisserman. “Deep Face Recognition”. In: *BMVC*. 2015.

[31] F. Rosenblatt. “The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain”. In: *Psychological Review* (1958).

[32] Fabrizio Rossi et al. “AI-driven personalization in retail: Challenges and opportunities”. In: *Journal of Retailing* (2021).

[33] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. “Learning representations by back-propagating errors”. In: *Nature* (1986).

[34] M. Sandler et al. “MobileNetV2: Inverted residuals and linear bottlenecks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[35] Connor Shorten and Taghi M Khoshgoftaar. “A survey on image data augmentation for deep learning”. In: *Journal of Big Data* (2019).

[36] Leslie N Smith. “Cyclical learning rates for training neural networks”. In: *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)* (2017).

[37] M. Tan and Q. Le. “EfficientNet: Rethinking model scaling for convolutional neural networks”. In: *Proceedings of the International Conference on Machine Learning (ICML)*. 2019.

[38] P. Viola and M. Jones. “Rapid Object Detection using a Boosted Cascade of Simple Features”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2001, pp. 511–518.

[39] Jing Wang and Yun Wang. “Age estimation using AAM and lifestyle factors”. In: *Pattern Recognition* (2015).
[40] Shenyan Wang et al. “Learning robust representations by projecting superficial statistics out”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).
[41] Wikipedia contributors. *Mean absolute error*. Accessed: 2025-01-14. 2023. URL: https://en.wikipedia.org/wiki/Mean_absolute_error.
[42] Cort J Willmott and Kenji Matsuura. “Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance”. In: *Climate Research* (2005).
[43] J. Yosinski et al. “How transferable are features in deep neural networks?” In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2014.
[44] Jichang Zhao, Yun Fang, et al. “Automatic age estimation using deep learning: A review”. In: *International Journal of Computer Vision* (2018).
[45] X. Zhu and D. Ramanan. “Face detection, pose estimation, and landmark localization in the wild”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.10 (2016), pp. 2105–2118.