THPoseLite, a Lightweight Neural Network for Detecting Pose in Thermal Images

Marcos Lupión, Vicente González-Ruiz, Javier Medina-Quero, Juan F. Sanjuan, Pilar M. Ortigosa.

***Abstract*—Nowadays, Smart Environments (SEs) enable the monitoring of people with physical disabilities by incorporating activity recognition. Thermal cameras are being incorporated as they preserve privacy. Some Deep Learning (DL) solutions use the pose of the users because it removes external noise. Although there are robust DL solutions in the visible spectrum, they fail in the thermal domain. Thus, we propose THPoseLite (Thermal Human Pose Lite), a Convolutional Neural Network (CNN) based on MobileNetV2 that extracts pose from Thermal Images (TIs). In a novel way, an auto-labeling approach has been developed. It includes a background removal using an optical flow estimator. It also integrates Blazepose (a pose estimator for Visible spectrum images (VSIs)) to obtain the poses in the pre-processed TIs. Results show that the pre-processing increases the percentage of detected poses by Blazepose from 19.55% to 76.85%. This allows the recording of Human Pose Estimation (HPE) datasets in the visible spectrum without requiring visible spectrum cameras or manually annotating datasets. Furthermore, THPoseLite has been embedded in an Internet of Things (IoT) device incorporating an edge Tensor Processing Unit (TPU) accelerator, which can process TIs recorded at 9 Frames Per Second (FPS) in real-time (12.28 FPS). It requires fewer than 6W of energy to run. It has been achieved using model quantization, decreasing the accuracy in estimating the poses by only 1%. The MSE of MobileNetV2 in test images is 35.48, obtaining accurate poses in 21% of the images that Blazepose is not able to detect any pose.**

***Index Terms*—Pose estimation, thermal image, auto-labeling, edge accelerator, quantization**

I. Introduction

THE progressively aging population is increasing the budget for human resources and infrastructures to deal with this situation [1]. However, only a small percentage of these people live independently in their homes without requiring the services of a caregiver or a retirement home [2]. In this context, careful home environment monitoring is critical to prevent serious health problems and risks, especially at older ages [3], [4]. For this reason, the use of SEs is aimed at solving several challenges: (1) adaptation to the person's capacities, (2) monitoring the person's health state and activities, and (3) triggering alarms when abnormal situations occur [5]. This monitoring process has been called Ambient Assisted Living (AAL) [6], where different sensors and devices are used to accomplish these conditions.

The first devices used in SEs, such as presence and pressure-on-surface detectors, were binary sensors that provided acceptable levels of accuracy in activity in a low invasive recognition way [7]. Later, the sensors were replaced by wearable devices, such as smartwatches and location tags, which obtained more accurate data about user movements, position and gestures [8].

The main limitation of wearable devices is the little battery autonomy [9] and their invasiveness [10]. Because of these limitations, Visible Spectrum (VS), usually Red, Green and Blue (RGB) cameras are incorporated into SEs. The captured images are processed using deep learning algorithms, achieving state-of-the-art (SOTA) results in tasks such as Activity Recognition (AR) [11] and fall detection [12]. However, a drawback of using this type of camera in SEs is that it leads to privacy issues, being considered as intrusive [13]. A solution to this is the use of low-resolution thermal cameras, as has been proposed in several works [14], [15]. These cameras record single-channel infrared (thermal) images containing the temperature of the objects and people. These images are processed similarly to VSI to perform the same types of tasks with similar accuracy [16], [17], but with the advantage of preserving people's privacy.

In DL solutions, the use of the poses of people is being widely adopted. In general, the abstraction provided by the use of poses allows the DL solutions: (1) to ignore the irrelevant features, such as background and person-specific characteristics, (2) to generalize better, and (3) to be trained more efficiently [18]. A body pose consists of several key points (also called landmarks) that indicate the localization of the joints representing the person's posture in the image. Several deep learning solutions that use the person's poses to perform fall detection [19] and AR [20] (among other tasks) have been proposed working in the VS domain.

HPE models in VSIs can obtain poses with a PCKh@0.5 value around 93 [21], [22]. However, their performance is usually far from good when the VSIs are replaced by thermal ones (less than 30% pose detection in [23] and 50% using [24] in several preliminary experiments with TIs). This happens because such systems were trained only with VSIs. For this reason, a collection of works specifically developed to work with TIs has been proposed in literature. They can be classified into two categories: (1) those that create an annotated dataset from scratch and re-train the HPE framework with these images [25], and (2) those that record paired VS and TIs and set the ground truth of the thermal pose as the pose obtained by a framework in the VS domain [26], [27]. The main problem of

Marcos Lupión, Vicente González-Ruiz, Juan F. Sanjuan and Pilar M. Ortigosa are with the Department of Informatics, University of Almería, CeIA3, Almería, 04120, Spain, (e-mail: marcoslupion@ual.es; vruiz@ual.es; jsanjuan@ual.es; ortigosa@ual.es)

Javier Medina-Quero is with the Department of Computer Engineering, Automation and Robotics Higher Technical School of Computer Engineering and Telecommunications, University of Granada, Granada, E-18071, Spain, (e-mail: javiermq@ugr.es)

the first approach is the high-demanding and error-prone work required to annotate a dataset. In the second approach, two cameras are needed, and the captured images must be aligned. Furthermore, in both cases, the solutions are not available to the community, and details about their architecture and training configuration are not provided.

When developing AAL solutions in SEs there are usually two main requirements [28]. First, the computation of these must be executed in real-time to react as quickly as possible to emergency situations such as falls [19], [29]. There exist HPE solutions in the VS that allow this, however, in the thermal domain, existing solutions do not run in real-time. Second, the devices have to run locally [30] to preserve the privacy of the users. These devices normally have low energy consumption and reduced prices, enabling the adoption of these in real environments.

In our work, we propose THPoseLite, a solution to the problem of HPE in TIs. It includes a novel automatic approach to labeling TIs, removing human error and the need for several devices. These are pre-processed, and a visible spectrum pose estimator is used to label them. Furthermore, a lightweight Neural Network (NN) is incorporated, building the first real-time HPE pose estimator using TIs in an IoT device. In the experimentation, we compare different pre-processing steps and NN architectures in a dataset recorded at the University of Almería. Furthermore, the execution of THPoseLite is assessed in the IoT device, analyzing HPE accuracy, inference speed and energy consumption.

Summarizing, the main contributions of this work are:

1) to design and evaluate the performance of a lightweight deep NN, THPoseLite, which obtains poses from TIs.
2) the auto-labeling images (obtained with a FLIR Lepton 3.5 camera) that are pre-processed using Blazepose [23] (a pose estimator in VS RGB images).
3) the incorporation of a TPU accelerator, which provides a real-time inference of poses.
4) a dataset recorded in the Smart Home of the University of Almería has been created. It contains 40481 Portable Network Graphics (PNG) TIs from 6 users, recorded from two different angles.

The remainder of this article provides a detailed description of the proposed approach. In Section II, a review of related works and the state of the art of HPE frameworks on VS and TIs is presented. Section III presents the proposed framework and the EC device on which it is deployed. Section IV shows the evaluation of the method. Finally, Section V lists the main findings and possible future works.

II. Related Works

The incorporation of low-resolution thermal vision sensors in SEs is not usually found in real environments [31]. This is mainly due to the lack of public datasets and robust solutions. Visible spectrum solutions benefit from pre-trained models allowing the creation of robust and customized solutions to most of the problems [32]. Another negative point of TIs is the cost of the cameras [31], being higher than that of visible spectrum cameras [33]. However, the main benefit of thermal cameras is privacy preservation, as thermal, not light information is captured in the image. Thus, the lower the resolution, the higher the privacy preservation [31], [33]. VSIs are found to be intrusive [34]. Another good point about TIs is the ability to record images in no light conditions [26], allowing AAL solutions to work even in these cases.

In this work, the problem of HPE in TIs is addressed. In HPE, DL solutions have arrived to outperform traditional methods. Poses inferred from the user can have 2 Dimensional (2D) or 3 Dimensional (3D) coordinates. Solutions obtaining 2D coordinates are more accurate because it is easier to create large annotated datasets. 3D HPE is more challenging, collecting datasets in controlled lab environments, and facing the problem of occlusions.

In the VSs, there is a large number of solutions in literature, making it difficult to set a SOTA solution because there is no clear and fair comparison between them [22]. These have been mainly classified as top-down, and bottom-up [21]. On the one hand, the top-down approach first detects the person and obtains the landmarks on it; therefore, the running time depends on the number of people in the image. In this case, the poses are detected only if the people are detected. On the other hand, the bottom-up approach first locates the different landmarks in the image and then tries to "build" the pose(s) of one or more people in it. The main problem with this alternative is the difficulty of incorporating new types of landmarks. Moreover, the larger the number of landmarks the model can detect, the greater the running time required.

Among the bottom-up solutions, OpenPose [24] stands out because it was the first real-time multi-person system to jointly detect the human body, hand, facial, and foot landmarks (up to a total of 135) per image. OpenPose is based on a fine-tuned VGG-19 deep NN [35] that obtains the part affinity fields in the image, and then a greedy algorithm tries to link the different joints of the people. This approach outperforms other solutions such as Mask R-CNN [36] and Alpha-Pose [37], which follow a top-down approach. As for the inference time, the OpenPose framework has an FPS lower than 1 in the MPII [38] and COCO datasets [39] when it is run on an Intel i7 processor. OpenPose and Alpha-Pose are open-source solutions, providing pre-trained models together with software that users can use to detect poses in their images.

Another solution is BlazePose [23], which obtains 3D landmarks of a single person. BlazePose first detects the person using a lightweight face-detector NN based on the Vitruvian Man, and then, the landmarks are identified by another NN . BlazePose has been designed to be fast, achieving real-time performance on mobile terminals. This solution was trained on a fitness dataset [40], [41] that is not publicly available.

The vast majority of pose estimators developed to date work with RGB images taken in the VS. There exist no large and well-annotated available datasets containing single and multi-person images with annotated poses in TIs. Therefore, there is no SOTA solution in the thermal domain.

In order to overcome this, the first approach is using HPE solutions in the VS domain to label paired TIs. For example, ThermalPose [26] annotates TIs extracting features from OpenPose. A camera recording aligned thermal and

VSIs are required to record the images. Openpose extracts the landmarks in the VS image, and due to the alignment between images, the landmarks are the same in both images. Results show an encouraging performance outperforming VS pose estimators in dark environments. The same idea is implemented in [27]. In this work, VSIs and TIs are captured by different cameras, and a residual network is used to infer the landmarks. In [42], the authors propose a transfer learning approach to obtain face points in TIs from two paired thermal and VS annotated datasets. An in-bed HPE is built in work [43]. It proposes a self-supervised framework using an autoencoder trained with paired VS and TIs.

Alternatively, the second approach is the manual annotation of datasets. For example, the authors in [25] create a new dataset, developing an annotation tool based on a previous object detection tool. In [44], a new dataset is created using a 3D motion capture system in a controlled lab environment, together with reflective markers attached individually on 21 joints. It uses Openpose to extract 2D parameters.

These solutions have some drawbacks: i) the need for two cameras in the first approach, which can be costly, ii) the time-consuming task of annotating images in [25], and iii) the time-consuming and costly process in [44]. A common problem is the creation of small datasets not covering a wide variety of poses and people, building non-robust solutions. In addition, there are no clear evaluation metrics in the case of TI solutions, as there is no standard well-annotated dataset to use as a reference.

Regarding the computational hardware used to obtain those poses, in some works, such as in [45] and [46], the images are obtained by fixed cameras and sent to the cloud, where they are processed to obtain a pose. Using cloud resources, the elapsed time between the image capture and the HPE is high. Furthermore, these approaches send personal data out of the SE, and sensitive data can be exposed. Although some HPE solutions in the visible spectrum are able to obtain poses in real-time [23], none of the analyzed solutions in the thermal domain do. This is a key point when developing AAL solutions in SEs.

summarizing, visible spectrum solutions cannot be used as a solution to HPE on TIs since datasets used to train do not contain TIs so their performance in the VS is deficient. For this reason, some domain-specific solutions were created manually annotating datasets, or using visible spectrum solutions to annotate paired visible and thermal spectrum images. Nevertheless, these are time-consuming, costly, and not able to run in real-time using IoT and low energy-consuming devices.

III. Methodology

This section describes the proposed solution (THPoseLite) to the problem of HPE on TIs. First, the IoT device used to perform the HPE in the real environment is described. Second, BlazePose is introduced. Third, the steps to build THPoseLite are described. Fourth, the theoretical basis and algorithms implemented in the pre-processing step of TIs are described. Finally, the NN architecture incorporated into THPoseLite is exposed and analyzed.

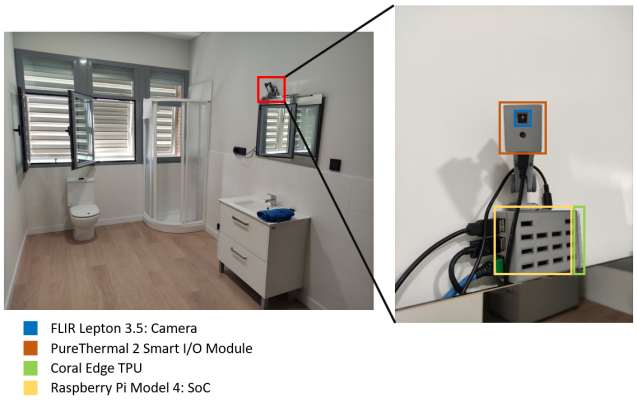

Fig. 1: Location of the IoT device and its components.

A. Embedded IoT Device

To overcome the drawback of using an external computer and sending the data outside sensitive environments, we present an EC system based on the IoT device developed under an Edge Computing approach in [47] that processes the images captured by an embedded camera and estimates the poses by running an NN in an accelerator. Thus, this device acts as a whole and can be installed anywhere. The total cost of the system is approximately €450. The main components of the IoT device are the following:

- **A Raspberry Pi (RPi) 4:** A well-known System on Chip is used in many IoT projects due to its low cost and communication capabilities. The processor is an ARM Cortex-172, with four cores working at 1.5 GHz. The RAM size is 4 GB. The price is roughly €40.
- **A PureThermal 2 Smart I/O Module + FLIR Lepton 3.5:**. With a price of €350, this device is an LWIR (Long-Wavelength Infra-Red) camera with radiometric capability, and it is smaller than a coin. This camera mounts a 160 x 120 active pixel focal plane array that captures the temperature of every pixel in the image. It has been integrated with the PureThermal 2 Smart I/O Module, a USB thermal webcam for the FLIR Lepton thermal imaging camera core. The PureThermal module has been pre-configured to work as a plug-and-play USB thermal webcam and integrated into an RPi through USB.
- **A USB Accelerator from Coral.AI:** This accelerator[1] incorporates an onboard TPU co-processor. Coral.AI can compute up to 4 trillion integer operations per second, using a small quantity of energy (less than 900 mA), and allows the user to significantly reduce the inference time of the NN compared to running it in the RPi 4. It costs approximately €60.

B. BlazePose

BlazePose is a lightweight (runs in real-time in current mobile devices) CNN that produces 33 2D body landmarks extracted from an RGB image of a person [48], [49]. It has been trained to find the pose in VSIs. Four values define each

[1] https://coral.ai/products/accelerator

landmark. Three of them are the 3D coordinates which locate it in the 3D space, and the fourth is a level of visibility which specifies the degree of visibility of the landmark in the image.

Unfortunately, BlazePose incorporates a face detector to segment the person [50], and if the face is not detected, it fails to estimate the pose. In our case, where the input images are thermal, the BlazePose network cannot detect the pose in most of the images because the face detector was not trained with TIs. However, there are cases where the pose is correctly obtained, and this knowledge can be transferred to a new NN to detect poses in images from the thermal domain. We use the set of detected poses to train our proposal, THPoseLite. Another problem we must face is that the images taken by the FLIR Lepton 3.5 camera have a low resolution and, depending on the camera's distance from the person, it is difficult to appreciate details such as the person's mouth, eyes, and expression. For these reasons, among the 33 landmarks given by BlazePose, the facial ones are removed from our system, resulting in 22 body landmarks.

C. THPoseLite

THPoseLite (our proposal, see Figure 2) has been implemented in a sequence of stages:

1) **Generation of the dataset**: a set sequence of inhabitants performing poses in the Smart Home of the University of Almería was recorded using the thermal camera. The details about the location and number of people can be found in Section IV-A.
2) **Pre-processing**: the TIs are pre-processed to: (1) remove the background, (2) extract the person from the image, and (3) center and scale it.
3) **Generation of the datasets using BlazePose**: hereafter, BlazePose processes the images and estimates the body poses. The images that contain the pose build the train and validation dataset. Those images where a body pose is not recognized are incorporated into a "No pose" dataset.
4) **Training of THPoseLite**: finally, THPoseLite is built in 32-bit floating-point precision format and trained using a training dataset. In addition, THPoseLite's model is quantized to integers of 8 bits and deployed in the Coral.AI accelerator to speed up the inference time of the poses using the IoT device.

D. Thermal Images pre-processing

The TIs are pre-processed to maximize the pose detection ratio success of BlazePose. This pre-processing consists of the following stages:

1) A reduction of the intensity of the background pixels of the images, considering that the background is formed by those areas of the images that experiment a motion estimation below a threshold.
2) After this, the background-attenuated images are binarized, resulting in a segmentation where each pixel of the image is classified as background or foreground (the person) (see Figure 4).
3) Finally, the person is located in the binarized image, where "noise"[2] is removed eliminating high-temperature areas in the background, and the person is centered in the image (see the Figure 5).

The main processing aspects are detailed below.

The images collected in our dataset show people in an environment with a wide variety of poses and fore-shortenings, time-varying temperatures, and different camera angles. As a result, the images integrate additional content in each sequence, allowing that the image's background can greatly impact the detection of a human pose.

In order to distinguish between foreground and background pixels (see Figure 3), we rely on the fact that the camera is static. Therefore, most of the background pixels (x, y) in the i-th image, $\mathbf{S}_i$, are also located at the coordinates (x, y) in the next $\mathbf{S}_{i+1}$ image. For this reason, we have estimated the motion at the pixel level (resulting in one motion vector[3] per pixel) between consecutive images using the Farnebäck dense optical flow estimator [51] provided by OpenCV[4].

The algorithm for attenuating the energy of the background in the images has been described in Figure 3. In this algorithm, the variable i (in Step 1) represents the index of the i-th image captured by the camera, and $\mathbf{B}$ (Step 2) is the background image, which initially is equal to the image $\mathbf{S}_0$. Then, iteratively, while the camera is capturing images (Step 3), we compute the background-attenuated image $\mathbf{E}$, by subtracting $\mathbf{B}$ from the next image in the input sequence (Step 3.(a)). The optical flow is computed and stored in the dense motion field $\mathbf{M}$ (Step 3.(b)). The classification between background and foreground pixels is stored in $\mathbf{G}$, where the foreground pixels are set to zero and the background pixels are copied from $\mathbf{S}_{i+1}$ (Step 3.(c)). Notice that the foreground is set to zero in order to avoid subtracting them in Step 3.(a). The background image $\mathbf{G}$ is updated with the information provided by $\mathbf{G}$, using a weighted moving average $\mathbf{G}$ (Step 3.(d)).

The result of the previous processing is an enhanced image sequence $\mathbf{E}$, whose pose detection success ratio is higher, where the intensities of the background pixels are decreased (see Figure 4). Notice that this technique works in real-time conditions when the background objects change their temperature smoothly and are quite resilient to fast-moving foregrounds (usually people), as long as the motion estimator has been properly configured. This implies that a large enough search area size is required, in the case of Farnebäck estimator, which is controlled by the number of levels l of the Gaussian pyramid and the window size w. In our estimator, both parameters remain constant over time, being $l = 2$ and $s = 16$.

In addition, it was appreciated that images containing a person-centered image led to increased performance in HPE using Blazepose. To locate the person in the image, input images are "binarized" (only two colors, black and white, are used). To compute the binary sequence $\mathbf{B}$, we have used Otsu's method [52], applied image-wise (see Figure 4). Next,

[2]From the training process perspective, that visual information that does not correspond to the person is considered noise.

[3]With sub-pixel accuracy (motion vector coordinates can indicate displacements smaller than the pixel size).

[4]https://opencv.org

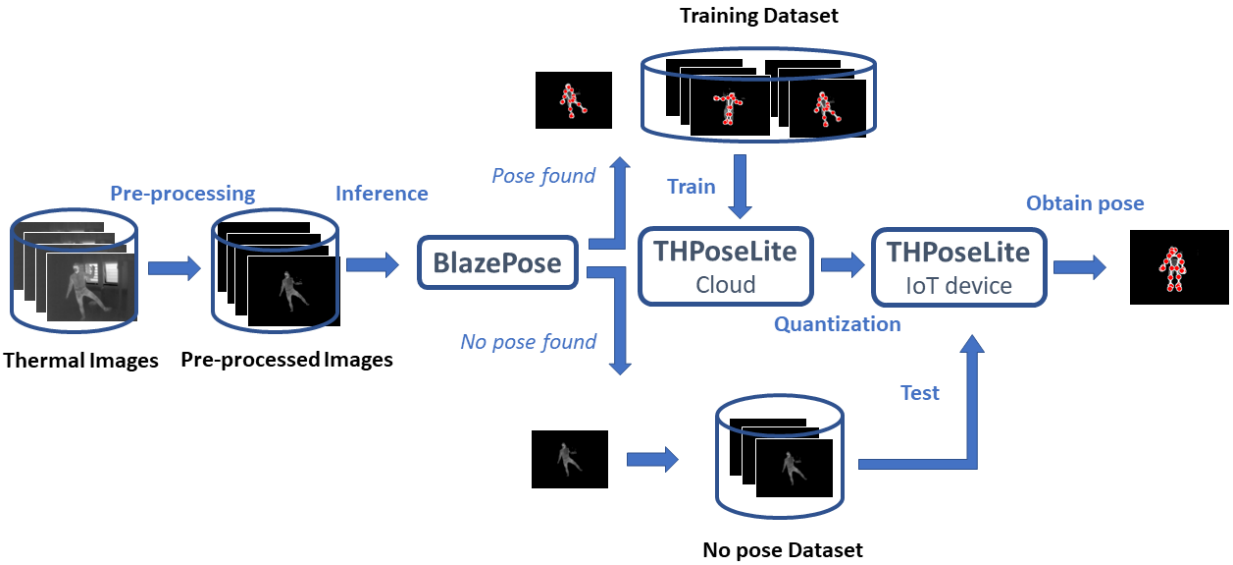

Fig. 2: Auto-Training of THPoseLite.

$\texttt{attenuate\_background}(S, l = 2, w = 16, \alpha = 0.99)$:

1) $i \leftarrow 0$
2) $\mathbf{B} \leftarrow \mathbf{S}_i$
3) Loop forever:
   a) Output $\mathbf{E}_i \leftarrow \mathbf{S}_{i+1} - \mathbf{B}$
   b) $\mathbf{M} \leftarrow \texttt{Farneback}(\mathbf{S}_i, \mathbf{S}_{i+1}, l, w)$
   c) $\mathbf{G}^{(x,y)} \leftarrow \begin{cases} \mathbf{S}_{i+1}^{(x,y)}, & \text{if } |\mathbf{M}^{(x,y)}| < 1 \\ 0, & \text{otherwise} \end{cases}$
   d) $\mathbf{B} \leftarrow \alpha\mathbf{B} + (1-\alpha)\mathbf{G}$
   e) $i \leftarrow i + 1$

Fig. 3: Procedure for decreasing the energy of the background in TIs.

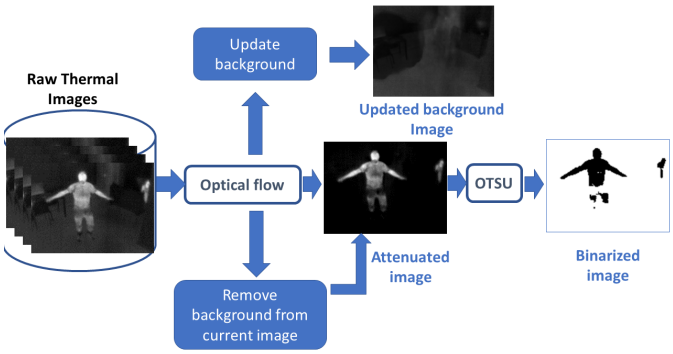

Fig. 4: First stage of pre-processing: Image segmentation.

a process to detect and center the person in the image was developed as depicted in Figure 5.

Small areas containing isolated points having an area below a threshold are eliminated as they contain some noise produced in the attenuation and binarization steps. Hereafter, areas having a medium size that are far from the big blobs (anything that is considered a large object or anything bright in a dark background, in this case, a person) are also eliminated. Those medium-sized areas close to the blobs can be parts of the person's body (such as the legs), so they are kept when their Euclidean distance is below a threshold.

Once the "noise" generated by the small and medium-sized areas has been removed, the blob is cropped from the image and centered in it. In our TIs, as there are different locations and perspectives, the people recorded have different sizes in each sequence. To generalize better and to detect more poses, these people are cropped from the original image, scaled, and situated in the center of the image. By doing this, all images have the same format, and Blazepose increases the success ratio R.

After the person-center processing, the original pixels involving the person in the image are incorporated into the

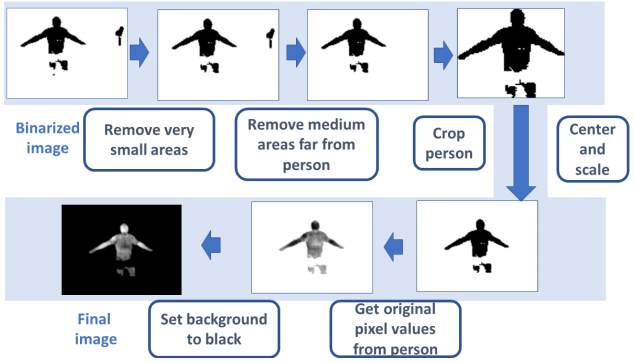

Fig. 5: Second stage of pre-processing: Person location and centering.

person's blob. This key process enables Blazepose to extract features from the person and to better identify the landmarks of the body parts. In addition, the color of the background is set to black to better highlight the person in the scene.

Figure 6 shows the attenuation of the background and the centering of the person in the image, and the different

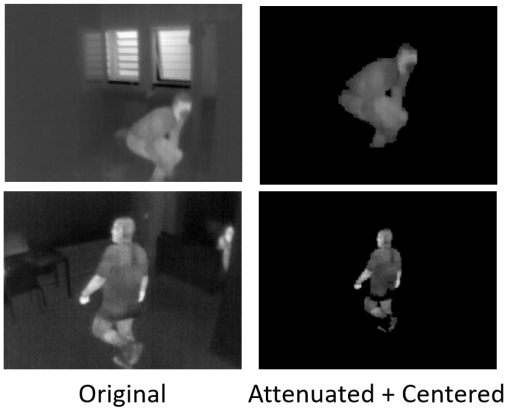

Fig. 6: Bathroom images with windows open.

backgrounds in the TIs can be appreciated. The camera's position was the same but rotated, and the perspective changed accordingly.

E. Neural network architecture

THPoseLite consists of an NN trained to detect poses in TIs. CNNs achieve SOTA results in some computer vision problems. Therefore, THPoseLite follows the structure of a CNN consisting of two main parts: the convolutional part and the classifier. The first part extracts spatial information from images and results in an embedded layer. This layer contains the information which allows the classifier to represent the key patterns of input images. In most cases, the classifier ends with a fully connected layer that maps the embedded layer to the NN 's output. In our problem, the NN 's output is not a classification of the input images in different classes. It consists of a regression problem where the coordinates and visibility of each of the key points of the person in the image have to be obtained.

In this work, the selected convolutional model is the MobileNetV2 NN . MobileNetV2 was designed to achieve SOTA imagery segmentation and object detection when executed on mobile devices. To fit in these, it has a reduced number of parameters. This NN incorporates point-wise and depth-wise convolutions that enable light filtering, reducing by a factor of 8 to 9 the time to execute the convolution operation. In addition, inverted residual blocks are added (bottlenecks). These blocks take as input a low-dimensional compressed representation which is first expanded to a higher dimension and then filtered with a depth-wise convolution, which allows us to keep the latent information from low layers until the end. We have incorporated MobileNetV2 into THPoseLite because it is a lightweight NN that runs on devices with low computing capabilities, such as the RPi .

The NN 's output is represented as a vector of 66 values representing the body pose. This consists of 22 landmarks, which save information about the user's joints. Each landmark is composed of three values: (1) the X coordinate, (2) the Y coordinate, and (3) the visibility of joint 7. Therefore, the second part of THPoseLite has to map the output of the convolutional part to 66 real values. Three fully convolutional

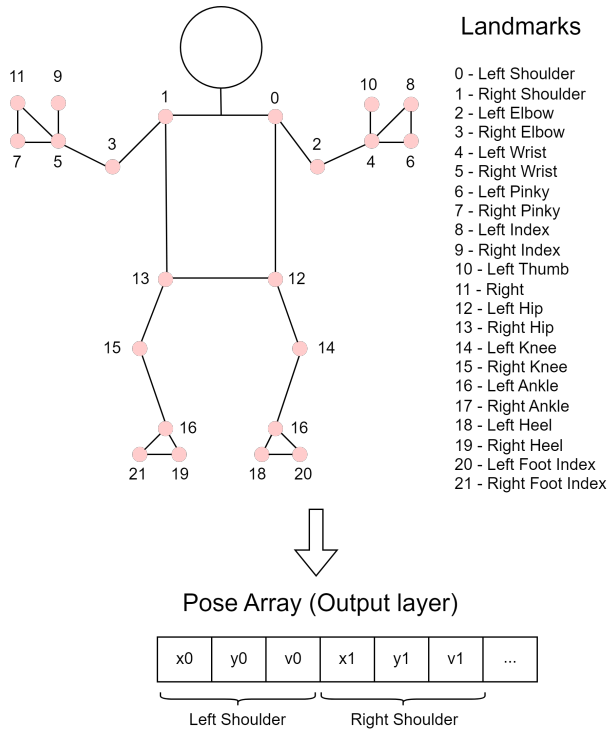

Fig. 7: Landmarks and mapping with the output layer of THPoseLite.

layers used in regression problems with 2048, 512, and 128 neurons, respectively are used for this task. It is important to remark that dropout layers have been incorporated between fully connected layers to prevent the NN from suffering overfitting.

In literature, MobileNetV2 has been trained with well-known VS image datasets (such as ImageNet). In deep learning solutions, it is very common to use pre-trained models to generalize better and speed up the training. This is known as transfer learning. However, after preliminary experimentation, transfer learning did not enhance the training of THPoseLite because of the different characteristics of the source domain of the images. Thus, training from scratch was required.

To sum up, Figure 8 shows the architecture of THPoseLite. Note that "Conv" represents a convolutional layer and "FC" a fully convolutional layer. The first part of the NN contains the MobileNetV2 architecture, incorporating convolutional layers together with bottleneck blocks (incorporating depth-wise convolutional layers). The output of this architecture is an embedding consisting of the most descriptive information from the image. After this, a fully connected regressor is incorporated, mapping the embedding from the MobileNetV2 architecture to the output layer.

IV. Results

This section includes the validation of the proposed HPE approach. First, the datasets and the configuration of the NN trainings are described. Second, the pre-processing techniques are compared in terms of MSE and MDE in order to set a configuration as the baseline for the rest of the experiments. Third,

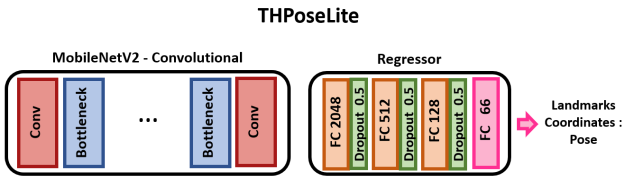

Fig. 8: Architecture of THPoseLite.

Device Used	Flir Lepton 3
Image dimensions	160×120
Number of images	40 481
Number of people	6
Number of locations	2
Conditions	Temperature: 20-25 ºC Morning-Afternoon

TABLE I: Dataset description.

three NN architectures are compared and the HPE results are presented in the proposed datasets. Finally, the resulting NN architecture is assessed in the IoT device, providing insights about the accuracy, inference speed, and energy consumption.

A. Dataset and training configuration

Before recording a custom dataset for collecting poses in a thermal domain, a literature review was conducted to look for datasets involving HPE in TIs. However, because we could not find such works or datasets, it was necessary to create a custom dataset to train and validate the proposed approach. Table I shows the main features of the created dataset.

The dataset was collected in the Smart Home of the University of Almería using two cameras located in the bathroom to record images from different perspectives. One camera was placed in front of the WC and the shower, and the other recording the hall. This was done because the inclusion of images from two different orientations helps to reduce the overfitting of the THPoseLite to a given background and environment. A total of 6 users were recorded in the dataset: two women and four men, with a height between 1.60 and 1.93 meters. In the recording set, users were instructed to behave normally but incorporate sudden movements such as raising their hands and legs, sitting on the WC, and squatting. Five users recorded at least two sequences of images with the first camera, and a long sequence involving all the users was recorded with the second camera. In the first perspective, windows were opened and closed to increase variability.

A total of 17 sequences (40481 images) were recorded. Sixteen were used to train and validate the proposed approach, and the remaining one was used to test it with unseen data. Sequences used to create the training and validation dataset were shuffled and split, where 70% of the images were incorporated in the training dataset and 30% of the images in the validation dataset.

In addition, several data augmentation techniques were applied in the training and validation dataset. The data augmentation factor was set to 2 in training and validation datasets. The data augmentation techniques were: (1) cropping, (2) rotating a random angle, and (3) flipping the image horizontally. In addition, before feeding THPoseLite with the training images, pixel values were scaled to the $[0, 1]$ range.

Summarizing, the datasets involved in this section are:

- **Training** (60738 images): This dataset contains the images and poses used to train THPoseLite. Two new images are obtained from each original image using data augmentation techniques to increase the variability in the dataset and to avoid overfitting. This data augmentation is necessary to increase the amount of data easily, as recording and annotating new data is difficult (each image hast 11 key points) and time-consuming.
- **Validation** (26034 images): This dataset contains the images and poses unseen in the training process to validate the model at the end of each epoch. It is very important that THPoseLite does not learn from these data in the training process. However, when the THPoseLite starts performing badly on this dataset, it shows that it is beginning to overfit the training dataset, so the training has to be stopped. In this work, data augmentation techniques were also incorporated into the validation dataset, having the same scaling factor as the training dataset.
- **Test** (2159 images): This dataset contains images and poses used to evaluate the THPoseLite with different images from those used in training and validation. As validation images are potentially similar to those incorporated in training, it is necessary to evaluate THPoseLite in a different domain. This case used a sequence containing 2151 images as a test dataset.
- **No pose** (9375 images): The auto-labeling tool analyzed all initial images to extract human poses in them and used this as the ground truth of each image. However, no pose was obtained in all images. Thus a set of images remained without having a ground truth pose. These images are not part of the training, validation, and test dataset but are incorporated into a new dataset. This dataset contains the poses where the original framework (BlazePose) could not detect any pose.

As for the training process, the TensorFlow Framework was used to implement the model and configure the training. The parameter configuration was the following:

1) The Adam optimizer was used, with a learning rate of 0.001.
2) The loss function considered was the Mean Squared Error (MSE).
3) The batch size was set to 16 images to speed up the training.
4) Finally, the total number of epochs needed to train THPoseLite was not fixed. As stopping criteria, when the MSE on the validation data does not improve the previous value after 30 epochs, the best parameters are stored in the system, and the training is finished. Therefore, the training process was unsupervised, and the number of epochs was adapted depending on the minimization of the MSE.
5) To speed up the training, the multi-GPU approach described in [53] was incorporated.

THPoseLite has been trained in an NVIDIA Tesla V100

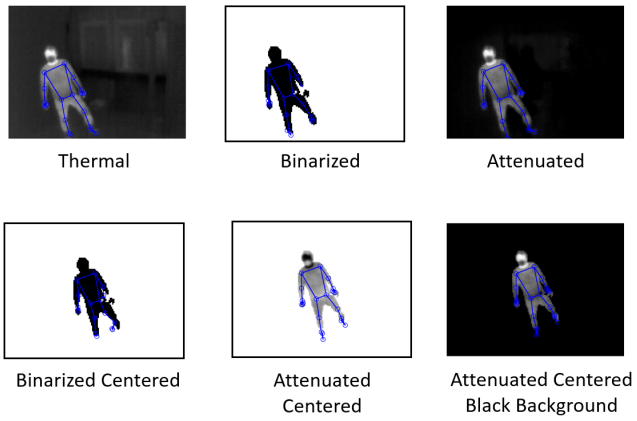

Fig. 9: Image pre-processing alternatives.

GPU using CUDA 11.0.2, hosted by a cluster node running CentOS 8.2 (OpenHPC 2), with 512 GB of DDR4 (3200 MHz) RAM, located at the University of Almería. However, notice that THPoseLite infers running in a Coral.AI accelerator attached to the RPi 4.

B. Pre-processing configurations

In this work, the pre-processing of the TIs was necessary to obtain poses using BlazePose. In the experimentation, six different types of images were compared to determine which type is better established as the input to BlazePose. These alternatives were:

- **Original thermal**: The raw thermal data which compose the image from the IoT device.
- **Attenuated**: Images having the background removed using the optical flow and the background recalculation.
- **Binarized**: Binary images obtained with the Otsu method applied to the attenuated images. The background is displayed in white, and the person is in black.
- **Binarized Centered**: Binary images containing the person in the center on it.
- **Attenuated Centered White Background**: Images containing the person-centered on it but keeping the raw value of the pixels which define the person. The background is white.
- **Attenuated Centered Black Background**: Images containing the person-centered, but keeping the original value of the pixels which define the person. The background is black.

The six different configurations are shown in Figure 9. As shown in Figure 9, the accuracy of the poses varies in the different configurations. Consequently, to choose the best alternative, they cannot only be evaluated using the number of images that produce a pose in BlazePose but also by considering the accuracy of the pose. For this reason, the error between configurations and ground truth is calculated. To measure the error, two metrics are calculated: (1) the MSE (Equation 1) and the Mean Distance Error (MDE) (Equation 2) between key points. MSE is incorporated because it considers each landmark's visibility value. MDE does not consider the visibility value but compares the location of the 22 landmarks in terms of pixels.

Image Type	Poses from total	MSE	MDE
Thermal	6,759 (19.55%)	32.16	5.04
Binarized	26,942 (66.55%)	140.82	14.01
Attenuated	15,025 (37.11%)	-	-

TABLE II: Blazepose's pose recognition with image pre-processing without centering the person.

Image Type	Poses from total	MSE	MDE
Binarized	27,921 (68.26%)	89.01	10.47
Attenuated	24,592 (60.73%)	42.15	6.18
Attenuated - Black Back.	31,083 (76.85%)	-	-

TABLE III: BlazePose's pose recognition with image pre-processing centering the person.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \tag{1}$$

$$\text{MDE} = \frac{1}{n}\sum_{i=1}^{n}\sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2} \tag{2}$$

In Table II, the configurations that do not center the person are compared. As can be seen, the Thermal image configuration obtains fewer poses (only in 19.55% of the images is a pose obtained), so the creation of a dataset only with these images would require a greater number of images. Comparing Binarized and Attenuated configurations, attenuated images allow obtaining more poses and lower error. The main difference between Binarized and Attenuated is that attenuated images keep information about joints that binary remove. This information helps the pose estimator to locate the joints better.

In addition to the experiment in the current dataset, the pre-processing approaches were compared using the dataset of the work [27]. This work uses paired visible and thermal spectrum images, so the ground truth landmarks on the visible spectrum are available. In [27], BlazePose obtained pose in 25% of the TIs, while the attenuated version obtained pose in 32.9% of the cases. Furthermore, the accuracy of the landmarks given by the thermal and attenuated images was compared to the landmarks in the visible spectrum. Results indicate that the Root Mean Squared Error (RMSE) of TIs concerning the thermal images is 5.47, while the error using attenuated is 5.28. Therefore, the use of poses obtained in the attenuated version is more accurate than the pose obtained using thermal images.

Table III compares the configurations that center the person. The configuration obtaining more poses is Attenuated with a black background (76.85%). Furthermore, it is the configuration getting more accurate poses. Binarized and Attenuated with white background configurations obtain more than 60% of poses, outperforming the configurations shown in Table II, not centering the person in the image. Here, the same conclusion can be extracted: Attenuated configuration obtains better poses than Binarized.

Finally, we highlight the performance improvement in that images with black backgrounds and centered person process-

ing allow BlazePose to increase the recognition rate. Thus, THPoseLite's input images have this configuration.

C. Human Pose estimation

THPoseLite incorporates MobileNetV2 as the convolutional part. However, two other architectures were also compared:

- **ResNet50** [54]: This NN was proposed to minimize the vanishing gradient problem in very deep NNs. This was achieved by incorporating skipping connections between layers. It contains 50 layers, including convolutional, batch normalization, max, and average pooling. These layers are grouped in residual blocks. These blocks are composed of two sets of convolutional, batch normalization, and Rectified Linear Unit (ReLU) layers with a skip connection between the input and the output. Our work incorporates a dropout layer after each residual block to avoid overfitting. This practice is very common in big NNs . Preliminary experimentation confirms the use of dropout in THPoseLite. It is appropriate to incorporate ResNet50 in this case because, due to its depth, features from the TIs can be captured better, enhancing the key points estimation.
- **U-NET** [55]: This NN was proposed to perform image segmentation tasks. It comprises an encoder and a decoder architecture using convolutional and batch normalization layers. Between the encoder and decoder, there is a latent space that contains the essential information of the input images to perform the segmentation. In addition, between the encoder and decoder layers, there exist skipping connections. These connections bypass one or more layers of the network and connect the network's input directly to the output. This allows the network to retain information from the input in the output image. Therefore, the output image can maintain the structural information of the input image, helping to predict the location of the key points and their visibility.

The three convolutional architectures compared: MobileNetV2, ResNet50 and U-NET, are applied in many computer vision works. In our problem, they were selected because, a priori, they could lead to good results.

The proposed convolutional architectures were evaluated with the validation and test datasets. Table IV shows the error from the ground truth regarding the MSE. In addition, the inference time in the IoT device and the number of parameters are displayed.

On the one hand, MobileNetV2 and ResNet50 have similar performance over the validation and test datasets. On the other hand, UNET behaves poorly in the validation dataset but improves in the test dataset. For both datasets, it is the worst NN.

In terms of inference time in the IoT device, MobileNetV2 outperforms ResNet50 and UNET because it was developed to be run on mobile devices and requires limited computing resources. It is also important to remark that MobileNetV2 has performed better than ResNet50 and UNET, having only 10.18% of the parameters of ResNet50 and 4.94% of the parameters of UNET. This number of parameters allows MobileNetV2 to have an excellent inference speed.

N. Network	Val.	Test	Inf. Time (ms)	Num. Parameters
MobileNetV2	37.96	36.57	74.8	4,138,626
ResNet50	39.08	35,36	483.8	40,625,934
UNET	53.60	40.78	664.8	83,722,691
Blazepose	-	-	815	-

TABLE IV: Comparison of the proposed NNs

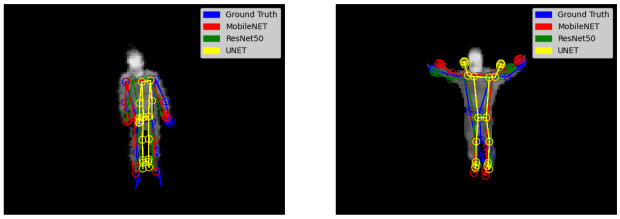

Fig. 10: Pose Estimation on validation dataset.

In addition, MobileNetV2 was compared to the ResNet50 architecture incorporated in the work [27]. MobileNetV2 achieved an RMSE value of 5.02 in the validation dataset, while ResNet50 had a value of 9.8.

Figure 10 shows the pose estimation of the proposed NNs in the validation dataset. MobileNetV2 and ResNet50 can infer the pose of the person correctly. However, UNET cannot. Analyzing the images in the validation dataset, UNET tends to generate a pose in the middle of the image, having short arms close to the body.

Figure 10 shows the pose estimation of the proposed neural networks in the test dataset. Considering that the test dataset contains images of a user with a different background and performing different poses, the results are considered encouraging in MobileNetV2 and ResNet50 for this complex and challenging problem.

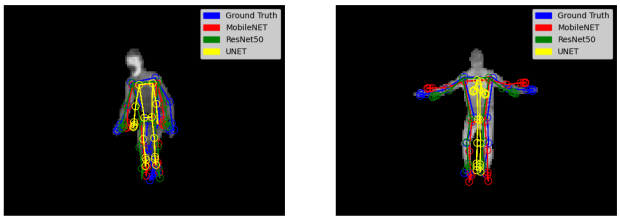

Fig. 11: Pose Estimation on test dataset.

Figure 12 shows the pose obtained in images where Blazepose did not recognize any pose. We note that the MobileNetV2 neural network is able to infer the pose in the displayed images. However, ResNet50 can predict a pose, but the result is incorrect. UNET is not able to predict any pose.

Finally, notice that THPoseLite is evaluated on images in which BlazePose has not obtained a pose. No reference is available to decide whether the estimated pose is correct. Therefore, several external users were asked to validate the system. The users must assess that the pose identified by

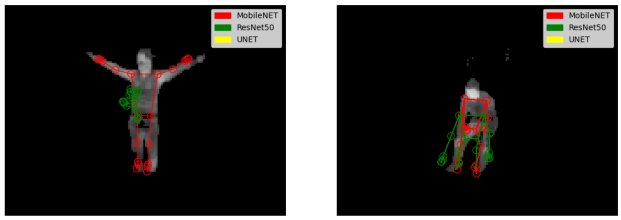

Fig. 12: Pose Estimation on images where Blazepose does not obtain pose.

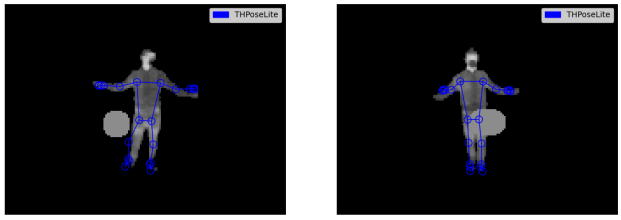

Fig. 13: Pose Estimation on images containing synthetic noise.

THPoseLite describes the user's posture. From a total of 9,375 images, 1,000 of them were randomly selected.

After evaluation, a correct pose recognition rate of 21% was obtained concerning the total number of images in which BlazePose has not obtained a pose. This result was promising, considering that the training images and poses used to train are obtained from BlazePose. This result showed room for improvement, incorporating new postures into the training dataset.

In the previous experiments, THPoseLite was evaluated with images containing a single person, achieving high accuracy. However, in smart environments, there can be some pets or household appliances that emit heat and therefore, disturb the TI. In the case of household appliances, they are static so the pre-processing incorporating the motion estimation removes these from the image. In the case of pets moving in the scene, the pre-processing step is not able to remove them from the image, so THPoseLite can be affected by this fact. In order to assess the behavior in these cases, some synthetic occlusions have been incorporated into the test dataset. Figure 13 shows that THposeLite is able to handle this noise and predict the pose accurately. However, it can be appreciated that a few artifacts are introduced in the HPE, for example, when the noise is not overlapping the person, the pose tends to be displaced by the noise. This could be removed in future works by incorporating these synthetic images in the training of THPoseLite in order to make THPoseLite resistant to external interferences such as moving objects or adversarial samples.

The source code that implements all the NN architectures is available at GitHub[5]. In addition, the trained TensorFlow models using 32-bit floating-point precision arithmetic, the TensorFlow Lite models, and the integer 8-bit quantized version are provided in a public Google Drive folder[6].

D. Performance on the IoT device

This work develops an NN that obtains poses from users in the thermal domain. In the experimentation, the model was built using the convolutional part of Resnet50, MobileNetV2, and UNET (as indicated in Section III-E). The resulting NNs differ in terms of HPE and inference time accuracy.

In our work, a non-functional requirement is that an NN model is incorporated into an IoT device (RPi 4) with low computational capabilities. On the one hand, smaller NNs will run more quickly and can process more FPS than NNs with a high number of parameters. On the other hand, small NNs tend to be less intelligent and produce worse results, so there is a trade-off between the speed of inference and accuracy.

Nowadays, the wall-time of TensorFlow NNs in devices such as RPi can be decreased using neural accelerators that are optimized to efficiently perform billions of operations per second, which is required in deep learning models. The accelerator incorporated in the IoT device uses an NN model in TensorFlow Lite format. When converting a model from TensorFlow to the TensorFlowLite format, no loss in precision is incorporated, as weights are stored in 32-bit floating point precision. However, the accelerator does not support 32-bit operations, and the model was required to be quantized in 8 bits. This quantization of the input values, weights, biases, activation functions, and obviously output values to 8 bits[7] has several advantages:

- The quantized model requires less memory. This allows it to be running in an Arduino.
- The inference time is reduced by a factor of 4 approximately. This minmizes the latency of the system.
- The inference can be performed in a dedicated accelerator, such as Coral.AI.
- The battery time (in the case of using it) increases for the same computational load.
- Compatibility with some accelerator. Some accelerators perform operations with a low-precision format, such as integers with 8 bits.

The main drawback generated by the quantization of the model is the precision loss in the operations and, therefore, in the final output of the system.

To quantize the model, there are different alternatives:

1) **Quantizing after normal training**: The NN is trained using 32-bit precision and then quantized by considering a validation dataset. The model inputs, outputs, weights, and biases are quantized, considering their maximum and minimum values when forwarding the validation dataset. In this stage, selecting the images to incorporate into the quantization process is key because the model is quantized accordingly.
2) **Quantization-aware**: In this case, the model is trained from scratch, simulating the quantization in the forward

[5]https://github.com/marcoslupion/THPoseLite.

[6]https://drive.google.com/drive/folders/1BSizNKKjcjbCyk_uvri4VKW0neZ7fsxE.

[7]Originally, weights and biases are represented with floats of 32 bits.

	Validation		Test		Inference Time	
Neural Network	MSE	Precision Loss	MSE	Precision Loss	Quantization (ms)	Speed Up
MobileNetV2	39.14	1.18	35.48	1.09	52.2 (19.15 FPS)	1.43
ResNet50	119.63	80.55	83.14	47.78	516.7 (1.93 FPS)	0.93
UNET	57.46	3.86	43.11	2.33	228.7 (4.37 FPS)	2.90

TABLE V: Comparison of the 32-bit precision and quantized NNs in terms of accuracy and inference time.

pass and updating weights using 32-bit precision. The resulting model is a full-precision model that is supposed to behave better when quantized. However, after the model's training, it is required to be quantized, and typically, a validation dataset is used to accomplish this task.

In this work, the first quantization approach was followed due to its good performance and ease of integration.

1) Pose estimation performance: To measure the loss in accuracy concerning the 32-bit precision model and the inference time in the IoT device using the Coral.AI accelerator, the different NN architectures proposed in Section IV-C were quantized and executed in the mentioned IoT device. Table V shows the obtained MSE in the validation and test datasets, with the loss in accuracy, inference time, and speed-up for the 32-bit precision model.

As can be seen, MobileNetV2 is the architecture having the best performance after quantization (39.14 and 35.48 MSE in validation and test datasets). This happens because its architecture was developed to perform well in mobile devices, incorporating fewer parameters than more complex NNs. The quantization process introduces some loss of precision every time a model parameter has been quantized. Thus, quantizing fewer parameters will reduce this loss. However, ResNET and UNET contain more parameters, and their architectures are deeper. Therefore, although it can be an advantage in terms of accuracy in complex datasets, quantizing these models produces a higher performance loss, making them unfeasible to run in neural accelerators. ResNet has a loss in MSE of 80% and 47.78% in the validation and test dataset. UNET suffers aloss in precision of 3.86% in validation and 2.33% in the test dataset.

Figures 14 and 15 show the performance of the quantized models in the validation and test datasets. The bad behavior of ResNet50 can be appreciated, while UNET and MobileNetV2 obtain similar results.

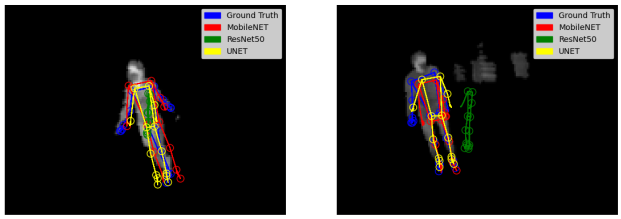

Fig. 14: Comparison between quantized neural networks in validation dataset.

Figures 16, 17 and 18, show the loss of precision of MobileNetV2 after quantization. It can be observed that the difference is not noticeable to the human eye.

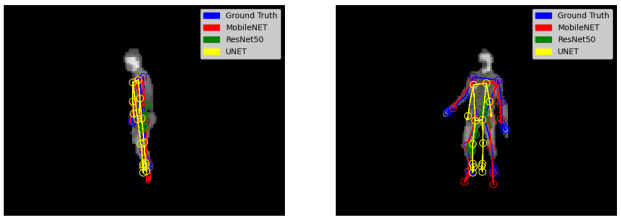

Fig. 15: Comparison between quantized neural networks in test dataset.

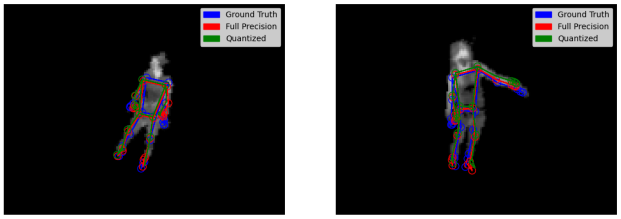

Fig. 16: Comparison between TensorFlow Model and Edge TPU model in the validation dataset.

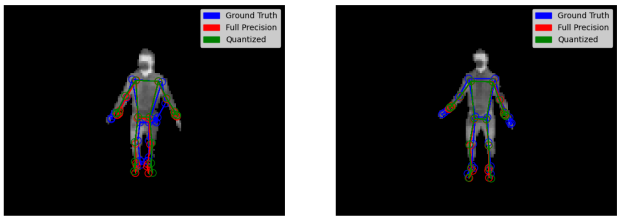

Fig. 17: Comparison between TensorFlow Model and Edge TPU model in the test dataset.

2) Inference time: In terms of inference time, the TPU accelerator allows MobileNetV2 and UNET to outperform the execution in the CPU. MobileNetV2 has a speed-up of 1.43, and UNET has a speed-up of 2.9. It is important to remark that this higher speed-up in the UNET architecture is due to the higher number of convolutional operations specially designed to be accelerated in accelerator devices. As for ResNet50, its inference time is increased using the accelerator. This difference in time is because the TPU is specially designed to accelerate Multiply ACcumulate (MAC) operations. Only this specific operation can be done amazingly fast on a TPU . All

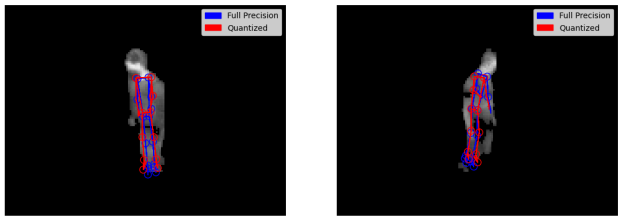

Fig. 18: Comparison between TensorFlow Model and Edge TPU model with images that Blazepose is not able to detect pose.

other operations, like loading weights, subtractions, additions, or dimension reduction, are time-consuming. For ResNet50, skip connections between layers are incorporated, performing addition operations between the input of a block of layers and its output. Therefore, it is concluded that MobileNetV2 is the most appropriate architecture to be used in THPoseLite, as it has the best accuracy when executed in the IoT device and a higher FPS value.

Finally, the pre-processing stage is evaluated in the target device. A sequence incorporated in the validation dataset is executed, and it was found that the average processing time per image is 29.2 ms. Therefore, the IoT device would need 29.2 ms to pre-process the image and 52.2 ms to execute the NN model. A total of 81.4 ms is required to process each image, resulting in 12.28 FPS . Considering that the FPS produced by the thermal camera is 9, THPoseLite enables real-time data processing in the proposed IoT device. Furthermore, THPoseLite has a speed-up of 15.61 compared to Blazepose.

3) Energy consumption: In addition to pose estimation results and inference time, the energy required by the IoT device has been measured. Thus, an ablation study was carried out to determine the role of each step in the pose estimation inference. A total of 6 configurations were executed:

- *Conf. 1*: Only capture images with the thermal camera.
- *Conf. 2*: Pre-Processing TI without running the model.
- *Conf. 3*: Pre-Processing TI and running the model on the Raspberry Pi processor.
- *Conf. 4*: Pre-Processing TI and running the model on Edge TPU.
- *Conf. 5*: Running the model on Edge TPU without pre-processing TI.
- *Conf. 6*: Running the model on Raspberry Pi processor without pre-processing TI.

Table VI shows the energy required by each of the configurations. Column *Instant Energy* shows the mean of the energy measured at each timestep. Column *Total Energy* shows the total energy spent when processing 300 images. On the one hand, the pre-processing of TI is the least energy-consuming step. On the other hand, configurations requiring the execution of the DL model are the most energy-consuming. The configuration using the Edge TPU device requires more instant energy to power the external device. However, in terms of total energy, compared to the configurations executing the model on the raspberry pi, it has lower energy consumption.

Config.	Pre-Processing	Model CPU	Model TPU	Instant Energy	Total Energy
Conf. 1	No	No	No	3.1	9.92
Conf. 2	Yes	No	No	3.8	42.56
Conf. 3	Yes	Yes	No	3.91	156.24
Conf. 4	Yes	No	Yes	5.22	133.47
Conf. 5	No	Yes	No	3.87	121.86
Conf. 6	No	No	Yes	5.15	103.77

TABLE VI: Energy consumption of different inference steps.

E. Limitations

As with the majority of studies, the design of the current study is subject to some limitations. Different scenarios and users will be tested in future work to study its generalization ability further.

In addition, while this study does not address the issue of data poisoning in HPE, it is an important consideration for future research [22]. In industrial environments using edge devices, the detection and correction of adversarial samples are essential to developing robust solutions, and some works propose some solutions to this [56]. Data poisoning refers to the intentional manipulation of training data to produce a biased or incorrect model [57]. This can have serious consequences on HPE, as it can lead to incorrect estimates and reduce the overall accuracy of the system. Further research is needed to address this issue and to develop robust methods for mitigating the impact of data poisoning on HPE [56].

V. Conclusions and Future Works

This work faces the problem of the HPE in TIs. First, an auto-labeling of the dataset has been developed, which is different from other works involving a tedious dataset construction process.

The pre-processing of the images lets Blazepose recognize poses in 78.65% of the cases. Only these images in which Blazepose recognizes the poses are considered for the dataset. This dataset comprises these original images and their corresponding poses. In addition, a data augmentation process is carried out to increase the variability in the dataset.

In the second part of the study, THPoseLite is proposed as a new CNN that extracts poses from TIs. MobileNetV2 is the selected architecture for the convolutional part of THPoseLite. MobileNetV2 has been compared to ResNet50 and UNET. After experimentation with validation, MobileNetV2 and ResNet50 are appropriate for recognizing poses in TIs.

In the third part of our work, THPoseLite is quantized in 8 bits to enable running in a TPU accelerator. Results show that the quantization process leads to a very small precision loss in MobileNetV2 but very high in ResNet50. Therefore, MobileNetV2 is found to be the most appropriate architecture for THPoseLite. Moreover, a speed-up of 15.61% is achieved concerning Blazepose. Thus, THPoseLite can be run in an edge device, processing TIs and obtaining poses in real-time (12.28 FPS , while the thermal camera records images at 9 FPS). Finally, the proposed approach is deployed in the Smart Home of the University of Almería.

In future works, other HPE frameworks will be tested with the pre-processed images to enhance the accuracy of

ground truth poses. Furthermore, the 3D HPE approach will be further studied, incorporating stereoscopic and depth cameras to annotate the datasets.

ACKNOWLEDGEMENTS

This work has been funded by the projects R+D+i PDC2022-133370-I00 and PID2021-123278OB-I00 from MCI-N/AEI/10.13039/501100011033/ and ERDF funds; by the Andalusian regional government through the project P18-RT-119, by the University of Almería through the project UAL18-TIC-A020, and by the Department of Informatics of the University of Almería. M. Lupión is a fellowship of the FPU program from the Spanish Ministry of Education (FPU19/02756). Lastly, this contribution has been supported by the Spanish Institute of Health ISCIII using the project DTS21-00047.

REFERENCES

[1] N. C. Hazra, C. Rudisill, and M. C. Gulliford, "Determinants of healthcare costs in the senior elderly: age, comorbidity, impairment, or proximity to death?" *The European Journal of Health Economics*, vol. 19, no. 6, pp. 831–842, 2018.

[2] M. Memon, S. R. Wagner, C. F. Pedersen, F. H. A. Beevi, and F. O. Hansen, "Ambient Assisted Living Healthcare Frameworks, Platforms, Standards, and Quality Attributes," *Sensors (Basel, Switzerland)*, vol. 14, no. 3, pp. 4312–4341, 2014.

[3] M. Terroso, N. Rosa, A. Torres Marques, and R. Simoes, "Physical consequences of falls in the elderly: a literature review from 1995 to 2010," *European Review of Aging and Physical Activity*, vol. 11, no. 1, pp. 51–59, 2014.

[4] World Health Organization, "Falls."

[5] P. Gharti, "A study of fall detection monitoring system for elderly people through iot and mobile based application devices in indoor environment," in *2020 5th International Conference on Innovative Technologies in Intelligent Systems and Industrial Applications (CITISIA)*, 2020, pp. 1–9.

[6] P. Rashidi and A. Mihailidis, "A Survey on Ambient-Assisted Living Tools for Older Adults," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 3, pp. 579–590, 2013.

[7] R. A. Hamad, A. S. Hidalgo, M.-R. Bouguelia, M. E. Estevez, and J. M. Quero, "Efficient Activity Recognition in Smart Homes Using Delayed Fuzzy Temporal Windows on Binary Sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 2, pp. 387–395, 2020.

[8] M. Lupión, J. Medina-Quero, J. F. Sanjuan, and P. M. Ortigosa, "DOLARS, a Distributed On-Line Activity Recognition System by Means of Heterogeneous Sensors in Real-Life Deployments—A Case Study in the Smart Lab of The University of Almería," *Sensors*, vol. 21, no. 2, 2021.

[9] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[10] S. Gaglio, G. L. Re, and M. Morana, "Human Activity Recognition Process Using 3-D Posture Data," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 5, pp. 586–597, 2015.

[11] I. Rodríguez-Moreno, J. M. Martínez-Otzeta, B. Sierra, I. Rodriguez, and E. Jauregi, "Video Activity Recognition: State-of-the-Art," *Sensors*, vol. 19, no. 14, 2019.

[12] M. M. Islam, O. Tayan, M. R. Islam, M. S. Islam, S. Nooruddin, M. Nomani Kabir, and M. R. Islam, "Deep Learning Based Systems Developed for Fall Detection: A Review," *IEEE Access*, vol. 8, pp. 166 117–166 137, 2020.

[13] J. Rafferty, J. Synnott, and C. Nugent, "A hybrid rule and machine learning based generic alerting platform for smart environments," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2016, pp. 5405–5408.

[14] C. Zhong, W. W. Y. Ng, S. Zhang, C. D. Nugent, C. Shewell, and J. Medina-Quero, "Multi-Occupancy Fall Detection Using Non-Invasive Thermal Vision Sensor," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5377–5388, 2021.

[15] S. A. Manssor, Z. Ren, R. Huang, and S. Sun, "Human activity recognition in thermal infrared imaging based on deep recurrent neural networks," in *2021 14th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*.

[16] B. Fu, N. Damer, F. Kirchbuchner, and A. Kuijper, "Sensing Technology for Human Activity Recognition: A Comprehensive Survey," *IEEE Access*, vol. 8, pp. 83 791–83 820, 2020.

[17] C. Zhong, W. W. Y. Ng, S. Zhang, C. D. Nugent, C. Shewell, and J. Medina-Quero, "Multi-Occupancy Fall Detection Using Non-Invasive Thermal Vision Sensor," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5377–5388, 2021.

[18] L. Song, G. Yu, J. Yuan, and Z. Liu, "Human pose estimation and its application to action recognition: A survey," *Journal of Visual Communication and Image Representation*, vol. 76, p. 103055, 2021.

[19] H. Ramirez, S. A. Velastin, I. Meza, E. Fabregas, D. Makris, and G. Farias, "Fall Detection and Activity Recognition Using Human Skeleton Features," *IEEE Access*, vol. 9, pp. 33 532–33 542, 2021.

[20] G. Batchuluun, D. T. Nguyen, T. D. Pham, C. Park, and K. R. Park, "Action Recognition From Thermal Videos," *IEEE Access*, vol. 7, pp. 103 893–103 917, 2019.

[21] T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang, and C. Yang, "The Progress of Human Pose Estimation: A Survey and Taxonomy of Models Applied in 2D Human Pose Estimation," *IEEE Access*, vol. 8, pp. 133 330–133 348, 2020.

[22] C. Zheng, W. Wu, C. Chen, T. Yang, S. Zhu, J. Shen, N. Kehtarnavaz, and M. Shah, "Deep learning-based human pose estimation: A survey," *arXiv preprint arXiv:2012.13392*, 2020.

[23] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, "blazepose: On-device real-time body pose tracking."

[24] Z. Cao, G. Hidalgo, T. Simon, S. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," *IEEE Transactions on Pattern Analysis amp; Machine Intelligence*, vol. 43, no. 01, pp. 172–186, 2021.

[25] R. Mehra, M. Chetty, and J. Kamalu, "Multiperson pose estimation using thermal and depth modalities," *Dept. Comput. Sci., Stanford Univ., Standford, CA, USA, Tech. Rep*, vol. 1, 2017.

[26] I.-C. Chen, C.-J. Wang, C.-K. Wen, and S.-J. Tzou, "Multi-Person Pose Estimation Using Thermal Images," *IEEE Access*, vol. 8, pp. 174 964–174 971, 2020.

[27] A. Polo-Rodríguez, M. Lupión, P. M. Ortigosa, and J. Medina-Quero, "Estimating Frontal Body Landmarks from Thermal Sensors Using Residual Neural Networks," in *International Work-Conference on Bioinformatics and Biomedical Engineering*. Springer, 2022, pp. 330–342.

[28] M. Hartmann, U. S. Hashmi, and A. Imran, "Edge computing in smart health care systems: Review, challenges, and research directions," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, p. e3710, 2022.

[29] M. Lupión, V. González-Ruiz, J. F. Sanjuan, J. Medina-Quero, and P. M. Ortigosa, "Detection of unconsciousness in falls using thermal vision sensors," in *Proceedings of the ICR'22 International Conference on Innovations in Computing Research*, K. Daimi and A. Al Sadoon, Eds. Cham: Springer International Publishing, 2022, pp. 3–12.

[30] K. Cao, Y. Liu, G. Meng, and Q. Sun, "An Overview on Edge Computing Research," *IEEE Access*, vol. 8, pp. 85 714–85 728, 2020.

[31] A. Naser, A. Lotfi, and J. Zhong, "Multiple thermal sensor array fusion toward enabling privacy-preserving human monitoring applications," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 16 677–16 688, 2022.

[32] S. Zhang, Z. Wei, J. Nie, L. Huang, S. Wang, and Z. Li, "A Review on Human Activity Recognition Using Vision-Based Method," *Journal of Healthcare Engineering*, vol. 2017, p. 3090343, Jul. 2017, publisher: Hindawi. [Online]. Available: https://doi.org/10.1155/2017/3090343

[33] L. USA, "VISIBLE vs. THERMAL DETECTION: Advantages and Disadvantages." [Online]. Available: https://www.lynred-usa.com/homepage/about-us/blog/visible-vs-thermal-detection-advantages-and-disadvantages.html?VISIBLE%20vs.%20THERMAL%20DETECTION:%20Advantages%20and%20Disadvantages

[34] A. Naser, A. Lotfi, and J. Zhong, "Adaptive thermal sensor array placement for human segmentation and occupancy estimation," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 1993–2002, 2021.

[35] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," 2014.

[36] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," 2017.

[37] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional Multi-person Pose Estimation," in *ICCV*, 2017.

[38] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, "A dataset for movie description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3202–3212.

[39] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," 2017.

[40] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D Human Pose Estimation: New Benchmark and State of the Art Analysis," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3686–3693.

[41] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 740–755.

[42] D. D. Poster, S. Hu, N. J. Short, B. S. Riggan, and N. M. Nasrabadi, "Visible-to-Thermal Transfer Learning for Facial Landmark Detection," *IEEE Access*, vol. 9, pp. 52 759–52 772, 2021.

[43] T. Cao, M. A. Armin, S. Denman, L. Petersson, and D. Ahmedt-Aristizabal, "In-Bed Human Pose Estimation from Unseen and Privacy-Preserving Image Domains," in *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022, pp. 1–5.

[44] Y. Zhu, W. Lu, R. Zhang, R. Wang, and D. Robbins, "Dual-channel cascade pose estimation network trained on infrared thermal image and groundtruth annotation for real-time gait measurement," *Medical Image Analysis*, vol. 79, p. 102435, 2022.

[45] Y. Muranaka, M. Al-Sada, and T. Nakajima, "A Home Appliance Control System with Hand Gesture based on Pose Estimation," in *2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)*, 2020, pp. 752–755.

[46] S. Pires, S. Rodrigues, L. B. Arokiadass, and S. Chopra, "A Real-Time Position Monitoring System For Fall Detection and Analysis Using Human Pose Estimation," in *2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*, 2021, pp. 1–7.

[47] M. Lupión, A. Polo-Rodríguez, J. Medina-Quero, J. F. Sanjuan, and P. M. Ortigosa, "On the limits of Conditional Generative Adversarial Neural Networks to reconstruct the identification of inhabitants from IoT low-resolution thermal sensors," *Expert Systems with Applications*, vol. 203, p. 117356, 2022.

[48] H. Xu, E. G. Bazavan, A. Zanfir, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "GHUM & GHUML: Generative 3D Human Shape and Articulated Pose Models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6184–6193.

[49] A. Zanfir, E. G. Bazavan, H. Xu, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "weakly supervised 3d human pose and shape reconstruction with normalizing flows," A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds.

[50] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, "BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs," 2019.

[51] G. Farnebäck, "Two-Frame Motion Estimation Based on Polynomial Expansion," in *Scandinavian conference on Image analysis*. Springer, 2003, pp. 363–370.

[52] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[53] M. Lupión, J. F. Sanjuan, and P. M. Ortigosa, "Using a Multi-GPU node to accelerate the training of Pix2Pix neural networks," *The Journal of Supercomputing*, vol. 78, no. 10, pp. 12 224–12 241, 2022.

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

[55] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[56] S. Shah, N. Jain, A. Sharma, and A. Jain, "On the robustness of human pose estimation," 2019.

[57] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Adversarial examples for neural networks," *arXiv preprint arXiv:1412.6572*, 2014.

VI. Biography Section

Marcos Lupión received his Master's degree in Technologies and Applications of Computer Science from the University of Almería, Spain, in 2020. He is currently a Ph.D. student at the same university. He receives a prestigious grant from the Ministry of Education, funding four years of Ph.D. research. His research interests include ambient intelligence, wearable devices, and deep learning solutions applied to eHealth.

Vicente González-Ruiz is an Associate Professor in the Informatics Department of the University of Almería. He received an M.Sc. degree in Computer Science from the University of Granada in 1992 and a Ph.D. in Computer Science from the University of Almería in 2000. He is a member of the Supercomputing-Algorithms Research Group of the University of Almería.

Javier Medina-Quero received his M.Sc. and Ph.D. degrees in Computer Science from the University of Granada, Spain, in 2007 and 2010, respectively. He is an Associate Professor of Computer Science at the University of Granada, Spain. He has published more than 35 articles in impact index journals related to Health and Computer Sciences. Furthermore, as a principal investigator and collaborating researcher, he has participated in several European, national, and regional research projects focused on e-health. He is also an Associate Editor of IEEE ACCESS and a Guest Editor in several JCR journals. His research interests include e-Health, Intelligent Systems, Fuzzy Logic, and Ubiquitous Computing.

Juan F. Sanjuan received his degree in Technical Engineering in electronic equipment from the University of Alcalá de Henares (Spain), a degree in Telecommunications Engineering from the Polytechnic University of Valencia and a Ph.D. in Computer Science from the University of Almería. He is currently a Doctor Contracted Professor in the area of Computer Architecture and Technology at the Department of Informatics (University of Almería). Nevertheless, since 2005 he has been a member of the 'Supercomputing-Algorithms, SAL' research group at the University of Almería. His research interests include the Internet of Things. Contact him at jsanjuan@ual.es.

Pilar M. Ortigosa is a Full Professor of Architecture and Computer Technology at the University of Almería, Spain. She received her M.Sc degrees in Physics and Electronic Engineering from the University of Granada in 1994 and 1996, respectively, and a Ph.D. in Computer Science from the University of Málaga in 1999. She is a member of the Supercomputing-Algorithms Research Group of the University of Almería. Her research focuses on High-Performance Computing, Metaheuristic Global Optimization, Computational Intelligence, Deep Learning, and the application to several real problems. Recently she has been working on the Internet of Things.