Fast heuristic method to detect people in frontal depth images

Carlos A. Luna[a], Cristina Losada-Gutiérrez[a,b], David Fuentes-Jiménez[a], Manuel Mazo[a],

[a] *Departament of Electronics, University of Alcalá, Ctra. Madrid-Barcelona, km. 33.600, Alcalá de Henares 28805, Spain.*

E-mails: carlos.luna@uah.es (C.A. Luna), cristina.losada@uah.es (C. Losada-Gutierrez), d.fuentes@edu.uah.es (D. Fuentes-Jimenez), manuel.mazo@uah.es (M. Mazo),

[b] Corresponding author. Tel: +34 918856906, fax: +34 918856591.

Abstract

This paper presents a new method for detecting people using only depth images captured by a camera in a frontal position. The approach is based on first detecting all the objects present in the scene and determining their average depth (distance to the camera). Next, for each object, a 3D Region of Interest (ROI) is processed around it in order to determine if the characteristics of the object correspond to the biometric characteristics of a human head. The results obtained using three public datasets captured by three depth sensors with different spatial resolutions and different operation principle (structured light, active stereo vision and Time of Flight) are presented. These results demonstrate that our method can run in realtime using a low-cost CPU platform with a high accuracy, being the processing times smaller than 1ms per frame for a 512x424 image resolution with a precision of 99.26 % and smaller than 4 ms per frame for a 1280x720 image resolution with a precision of 99.77 %.

*Keywords:*3D People detection, Depth camera, Frontal Depth images, Feature extraction, Head biometric classification.

1. Introduction

Robust human detection in images and videos has become a key task in computer vision due to its multiple applications in different areas such as human-robot interaction (Beyl et al., 2013; Muñoz-Salinas et al., 2005, 2007; Pereira et al., 2013), elderly people care (Ghiță et al., 2018; Solbach & Tsotsos, 2017; Tomoya et al., 2017) or security and video-surveillance (Dumoulin et al., 2018; Sumalan et al., 2018; Yang et al., 2016; Yimyam et al., 2018). Because of that, it has received great attention in the last years, existing numerous works in the literature that tackle people detection and counting using different sensors and approaches.

The first works in the literature were based on the use of RGB cameras for people detection (Jeong et al., 2013; Muñoz-Salinas et al., 2005, 2007; Ramanan et al., 2006; Wojek & Schiele, 2008). These works present good results in controlled conditions, but its performance drops significantly when there exist occlusions, poor lighting conditions or important lighting changes.

To reduce the effect of occlusions, there are different approaches in the literature that locates the camera in a top-view configuration (Sidla et al., 2006). Furthermore, in recent years, there have appeared the RGB-D cameras, such as Kinect, (Sell & O'Connor, 2014; Smisek et al., 2011) or the Asus Xtion pro (Migniot & Ababsa, 2013), which, in addition to color images, provide depth information (distance from each point to the camera). In this context, numerous works use these RGB-D cameras for people detection, tracking and counting (Del Pizzo et al., 2016; Liciotti et al., 2017; Zhang et al., 2012).

In some scenarios, systems that use color or RGB-D information cannot be used, due to legal issues related to privacy, since the information available in a color image allows recognizing the identity of people that appear in it. With the development of new depth sensors, based on Time of Flight (ToF) or structured light, which only provide depth data, different works have appeared where the detection of people is done using only these depth data (Bevilacqua et al., 2006; Fernandez-Rincon et al., 2017; Jia & Radke, 2014; Luna et al., 2016; Stahlschmidt et al., 2013; Wang et al., 2018) and thereby

preserving people's privacy. Moreover, the use of depth cameras reduces the effect of lighting changes, since they do not require an external lighting source.

Regardless of the technology used, among the main problems of optical depth sensors marketed, are the systematic and random errors that are made in depth measurements and the high number of invalid pixels present in maps or depth images. In many human detection works (Fernandez-Rincon et al., 2017; Luna et al., 2016; Stahlschmidt et al., 2013; Wang et al., 2018) a pre-processing step is used to smoothens the depth image and to estimate the value of the invalid pixels. This pre-processing carries out a high computational cost. One of the advantages of the method proposed in this work is that this pre-processing step is not required.

It is worth highlighting that most of the previously cited proposals (both RGB-D and depth based) use the camera in a top-view configuration to reduce occlusions. However, this configuration also reduces the study area to a small area under the camera, which highly depends on the camera location height. In order to increase the area of study, some works use depth cameras in a high frontal location (Tian et al., 2018). This configuration allows incrementing the area in which people can be detected, however, it also increases the effect of occlusions.

In recent years, the improvements in technology and the appearance of large-scale datasets have led to an increase in the number of works based on deep-learning for people detection (Fang et al., 2015; Fuentes-Jimenez et al., 2020), that obtain good accuracy. However, all these deep-learning-based approaches have a high computational cost that prevents their real time execution. Other authors (Da Silva Guizi & Kurashima, 2016; Khan et al., 2017; Sun et al., 2019), to avoid the stages of training and reduce processing times have used a template of parts of people (heads or heads and shoulders).

Despite the large number of works dealing with the people detection task, it is still an open and challenging issue on which the scientific community continues to work. In this context, this paper presents a real time and robust method for people detection using only depth images, captured with a camera in a high frontal position, based on biometric constrains. The proposal uses only depth information, allowing people's identity

preservation, while reducing lighting dependence. Furthermore, the camera location increases the area under study, but it also leads to an increase in the occlusions. The use of a 3D ROI (Region of Interest) reduces the effect of occlusions. Besides, the proposal has a very low computational cost that allows its real time execution, even in low-power systems, without needing a Graphical Processing Unit (GPU).

The rest of the paper is organized as follows, section 2 describes our proposal, then section 3 presents the results and makes a discussion of them, and finally, in section 4 the conclusions of the work are exposed.

2. Proposed solution

As it was commented in Section 1, the proposal for people detection only uses depth measurements captured by a camera located in frontal position. A general block scheme, including all the stages involved in the proposed method is shown in Figure 1. There are two different processes, an offline process and an online one. In the offline process, a set of background images is recorded, and the average value of that background is obtained. The online process includes four stages: in the first one a depth image is captured; then, the pixels corresponding to the background are removed. In the third stage, it is performed the detection of objects that could correspond to a person's head; and finally, in the fourth stage, it is carried out the discrimination between people and other objects, using biometric parameters of the people's head.

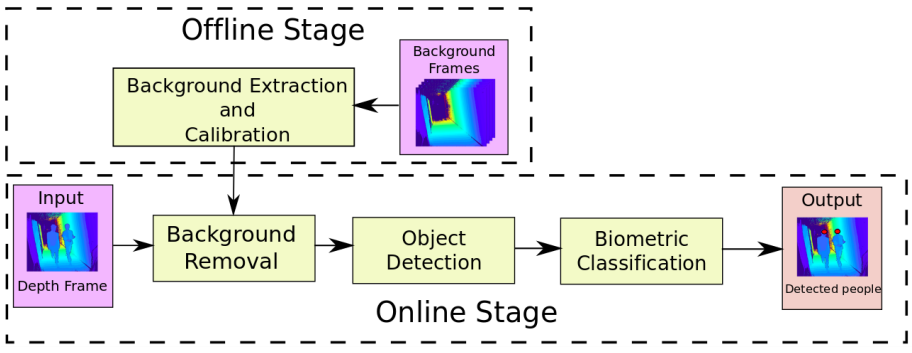


Figure 1. General block scheme of the proposed method to detect people.

2.1 Calibration and Background extraction.

Depth images provided by Time of Flight (ToF) sensors have several error sources related to their operating principles, such as the motion blur (Lee et al., 2012) caused by object and camera movement, the multipath effect (Jimenez et al., 2012), the limitation of the power of the IR illumination as well as the color and the type of material of the objects. Whereas that, in the sensors based on active stereo vision or on structured light, the main error causes are depth shadowing, IR dot splitting, spreading, and occlusions.

Since the noise level in depth measurements is high, and it increases proportionally with the distance between the objects and the sensor, it is necessary to extract the background from a set of N depth images. In the following, the depth images will be named as D. We assume that a pixel of the image D_i ($i = 1, 2, 3, ..., N$) is valid to contribute to the background depth image B if its value is between a minimum (d_{min}) and a maximum (d_{max}) depth. Empirically, we assume that if a pixel is valid in more than 20% of the N depth images, its value is taken as the average of all the values in which it has been valid. If the percentage is lower, it is assigned the value 0 that indicates that it is an invalid pixel. The values of d_{min} and d_{max} depend on the working depth of the used sensor.

The sensor is located in a frontal position, at a height higher than the maximum height of a person (h_{p_max}) and focused on the scene with a rotation angle α around the Y axis, as shown in Figure 2. As it can be seen the origin of the camera coordinate system (O_s) is displaced to a height (h_{sensor}) with respect to the origin (O) of the world coordinate system (X, Y, Z), which we will take as reference to determine the position of the objects or people in the scene. The values of tilt angle α and h_{sensor} are calculated through a calibration process.

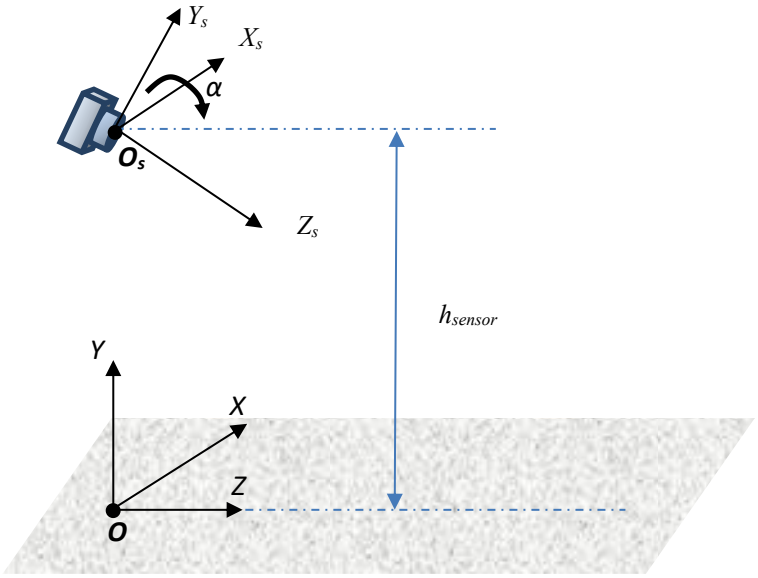


Figure 2. Camera location at a height h_{sensor} and considered coordinate systems.

The previously modeled background B is removed from each new depth image D. This process is done considering the following criteria that must be met for each pixel with coordinates (u, v) on the image plane, where we refer to the pixels with coordinates (u,v) as $B_{u,v}$ and $D_{u,v}$:

- $B_{u,v} > 0$, the pixel (u, v) of background image is valid and ~~if~~ its value is between d_{min} and d_{max} ($d_{max} > B_{u,v} > d_{min}$).
- $d_{max} > D_{u,v} > d_{min}$
- $| D_{u,v} - B_{u,v} | > d_{betw\_obj}$, where d_{betw_obj} is the minimum depth between two objects.

If the above conditions are fulfilled $D_{u,v}$ keeps its value, otherwise $D_{u,v} = 0$.

2.2 *Object detection.*

In order to detect people or other objects in the depth image D, we have developed an algorithm that is represented by means of the block diagram shown in Figure 3. The different stages of this algorithm are described below.

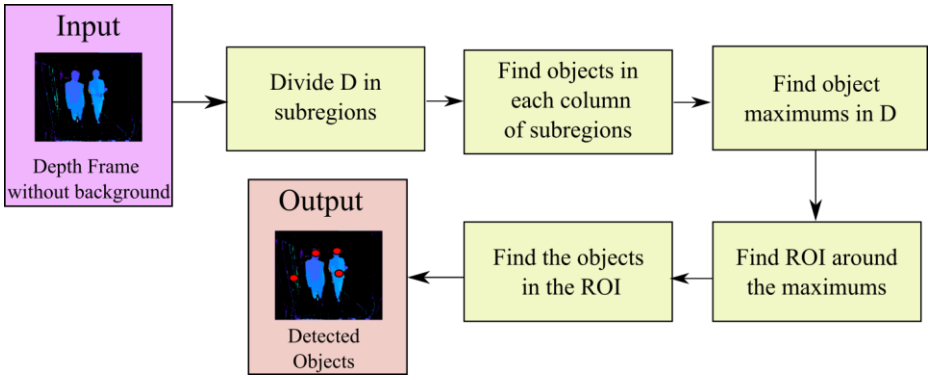


Figure 3. Block scheme of the object detection procedure.

1. ***Divide* D *in subregions* (SR).** In order to reduce the noise in the depth image and improve computational efficiency the image $D \in \mathbb{R}^{U \times V}$ is divided into CN x RN subregions of $N_{Px} \times N_{Px}$ pixels, see Figure 4:

$$CN = \frac{U}{N_{Px}};\ RN = \frac{V}{N_{Px}} \tag{1}$$

$$N_{Px} = f_x \frac{l}{d_{max}} \tag{2}$$

where f_x is the camera focal length divided by the pixel dimension (assuming square pixels), $l \times l$ is the minimum area at Z = d_{max} (in the optical center of the sensor). The value of l must be selected so that at the greatest working distance, the area of a head is covered by at least 4 subregions. Considering the dimensions of a head (Yoganandan et al., 2009) we chose $l = 65\ mm$. For the used sensor (Intel RealSense D435) with f_x= 644.45 and $d_{max} = 3000\ mm$, then N_{Px}»14

For each subregion $SR_{R,C}$ $(R = 1, \dots, RN, C = 1, \dots, CN)$ both, the average value $\bar{d}_{R,C}$ and the number of valid pixels $V_{R,C}$ are calculated. If the $V_{R,C}$ is less than 20% of the total number of pixels ($N_{Px} \times N_{Px}$), the $SR_{R,C}$ is discarded, assuming that valid pixels belong to an edge or can be noises.

Using the $\bar{d}_{R,C}$ and the calibration parameters, the coordinates (X, Y, Z) of the center of $SR_{R,C}$ respect to the origin of the world coordinate system (*O*) are calculated. In the following, they will be named as $Y_{R,C}$ and $Z_{R,C}$ respectively. If the $Y_{R,C} > h_{max}$ or $Y_{R,C} < h_{min}$, then the $SR_{R,C}$ is discarded, being h_{max} and h_{min} the maximum and minimum heights of people.

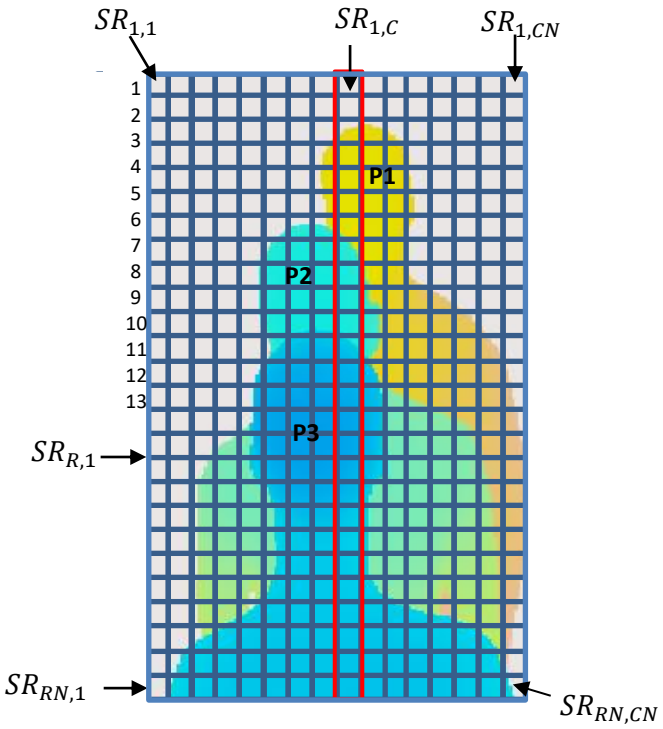


Figure 4. Example of the division of an image D in subregions with a fixed $\boldsymbol{N_{Px}}$.

2. ***Find the objects in each column of subregions*.** The SR belonging to the same object are grouped within each column. Grouping the SR in columns instead of grouping them by adjacency, avoids that when there exists an occlusion, parts of the same object can be considered as different objects. This can be seen in object P2 in the example of Figure 4, in which there is no continuity between the head and the shoulders in the depth image.

For each column C the number of objects N is obtained. To determine the subregions $SR_{R,C}$ that belong to an object within the column, starting from $SR_{1,C}$, R is incremented until the first valid $SR_{RI,C}$ is found ($h_{max} \geq Y_{R,C}$ and $Y_{R,C} \geq h_{min}$). In case of occlusion, this $SR_{RI,C}$ corresponds to the object that is furthest away. Once the initial $SR_{RI,C}$ is determined, R continues to be increased until it is satisfied that: $Z_{R-1,C} - Z_{R+1,C} > d_{betw\_obj}$. This means that $SR_{R-1,C}$ and $SR_{R+1,C}$ belong to two objects that are at different depths respect to O, as well as that $SR_{R,C}$ is the boundary between them. Therefore all the SR that are from $SR_{RI,C}$ to $SR_{R-1,C}$ belong to the same object and $SR_{R+1,C}$ is the first SR of the next one.

For each object~~,~~ the following features are obtained:

- RI_C. The first row R where a valid $SR_{R,C}$ of the object is located.
- NSR_C. The number of *SR* that belong to that object.
- $\bar{Z}_C$. The average value of the set of all depths of the subregions ($Z_{R,C}$) that belong to the object.

By the way of example, if we look at the column marked in red in Figure 4, the subregions $SR_{3,C}$, $SR_{4,C}$, $SR_{5,C}$ and $SR_{6,C}$ belong to the object P1, the $SR_{8,C}$, $SR_{9,C}$ and $SR_{10,C}$ correspond to the object P2**,** and the object P3 includes SRs from $SR_{12,C}$ to $SR_{RN,C}$ or until it is met that $Y_{R,C} < h_{min}$. The subregions $SR_{7,C}$, and $SR_{11,C}$ may belong to one of the edge objects.

3. ***Find object maximum in D.*** To determine the *SR* that may correspond to the maximum (Pi) of the object *i,* the following criteria are considered:

   a) For all C with $N > 0$, identifying as Pi the minimum row R_{min} (eq. (3)) associated with each C ($Pi_{R_{min,C}} = SR_{R_{min},C}$).

$$R_{min} = Min(\forall RI_C \in C) \tag{3}$$

   b) In order to remove small objects, assuming that the area of a head is covered by at least 4 subregions, Pi is discarded and eliminated if:

   - $NSR_c \leq 2$
   - At least one adjacent column ($C-1$ or $C+1$) does not have an object in the same distance range ($|\bar{Z}_C - \bar{Z}_{C+1}| < d_{betw\_obj}$) and with a $NSR_{c\pm1} \leq 2$

4. ***Find ROI around the maximum.*** In order to have more precise information about each object, a region of interest (ROI) is processed around each maximum found. The ROI dimensions depend on the depth of the object. Therefore, the average value of the depth ($\bar{d}_{R_{min},C}$) of all SR that are in the column of the maximum and in the adjacent columns is determined by eq. (4), whereas the mean value of Z respect to O is defined in eq.(5):

$$\bar{d}_{Pi} = \frac{\sum_{j=C-1}^{j=C+1} \sum_{R=RI_j}^{RI_j+NSR_j} \bar{d}_{R,j}}{\sum_{j=C-1}^{j=C+1} NSR_j} \tag{4}$$

$$\bar{Z}_{Pi} = \frac{\sum_{j=C-1}^{j=C+1} \sum_{R=RI_j}^{RI_j+NSR_j} Z_{R,j}}{\sum_{j=C-1}^{j=C+1} NSR_j} \tag{5}$$

For an area (in mm^2) of $l_{ROI} \times l_{ROI}$, the number of pixels within of the ROI can be computed as:

$$N_{Px\_ROI} = f_x \frac{l_{ROI}}{\bar{d}_{Pi}} \tag{6}$$

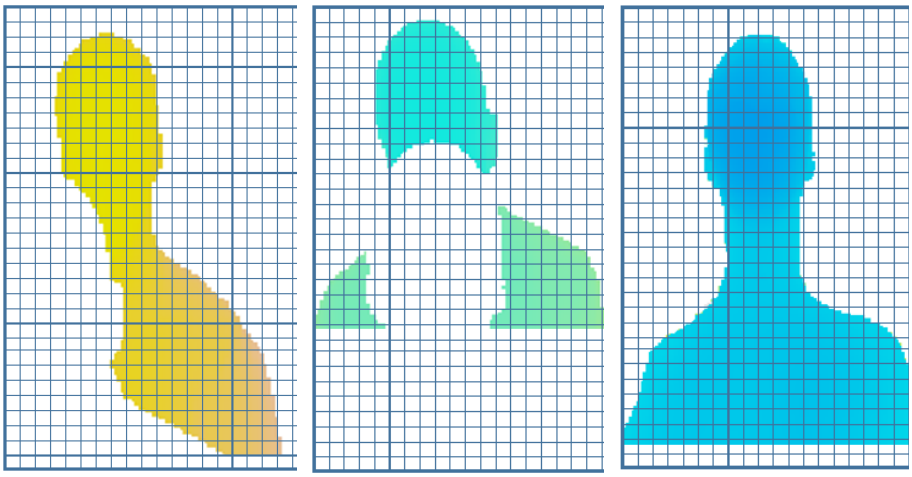
Since the top-left coordinates (u, v) of the maximum are ($R_{min} \times N_{Px}, C \times N_{Px}$), a zone of interest must be defined around it. In our case, taking into account the biometric characteristics of people and that these coordinates can have a deviation of tens of *mm*, we assume margins of 100 *mm* up, 500 *mm* down and 400 *mm* on both sides. The pixel coordinates of the top-left (u_{min}, v_{min}) and down-right (u_{max}, v_{max}) of the ROI are given by the following expressions:

$$\begin{aligned} v_{min} &= R_{min} \times N_{Px} - \frac{100\ mm}{l_{ROI}} \times N_{Px\_ROI} \\ v_{max} &= R_{min} \times N_{Px} + \frac{500\ mm}{l_{ROI}} \times N_{Px\_ROI} \\ u_{min} &= C \times N_{Px} - \frac{400\ mm}{l_{ROI}} \times N_{Px\_ROI} \\ u_{max} &= C \times N_{Px} + \frac{400\ mm}{l_{ROI}} \times N_{Px\_ROI} \end{aligned} \tag{7}$$

5. ***Find the objects in the ROI.*** Once the ROI has been defined, to increase the robustness of the detection of the object present in the ROI, the following procedure is followed:
   a) Discard all pixels that do not belong to the object: $ROI_{u,v} = 0$ if $\left|ROI_{u,v} - \bar{d}_{Pi}\right| > d_{betw\_obj}$. Figure 5 shows the ROI for each of the objects presented in Figure 4.
   b) The ROI is divided with a similar procedure to that described in the first step of the algorithm, where the number of pixels N_{Px_ROI} and the subregions are discarded also if:

$$\left|Z_{R\_ROI,C\_ROI} - \bar{d}_{Pi}\right| > d_{betw\_obj}$$

   c) To find the SR that corresponds to the maximum of the object ($Pi_{R\_ROI_{\min},C\_ROI}$), the procedures described in the steps 2 and 3 of this algorithm are applied to the ROI.



(a) (b) (c)

Figure 5. Division of ROI of the image D in subregions with a variable $\boldsymbol{N_{Px}}$. (a) P1, (b) P2 and (c) P3 of the figure 4.

2.3 Biometric classification

In this step, the classification of each of the objects ($Pi_{R\_ROI_{\min},C\_ROI}$), obtained in step 5.c), described in the previous section, is performed, to determine if it corresponds to a person's head. A general block diagram of the classification process is shown in Figure 6.

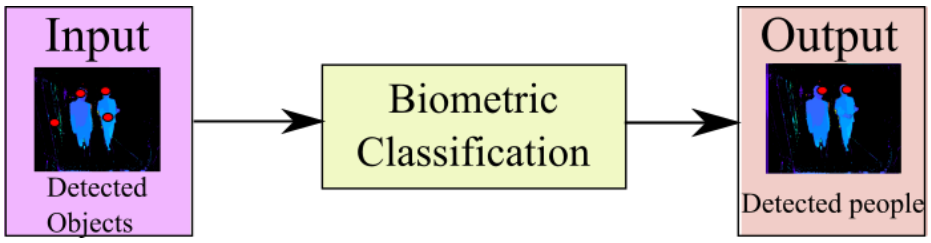


Figure 6. General block diagram of the biometric classification process.

Based on the biometric characteristics of people's heads (Yoganandan et al., 2009), we have used two criteria to discriminate between people and other objects that may be present in the scene.

The first criterion is based on the fact that the diameter of the head (*h_dia*) of an adult person, seen frontally or laterally, must be in a range between 120 *mm* and 340 *mm* (Yoganandan et al., 2009). To measure *h_dia* we use the head projection in the plane (X, Y), taking 100 *mm* below the center of the subregion where the maximum is located (eq. (8)), see Figure 7.

$$h_{dia} = \sqrt[2]{\left(X_{R_{side2},C_{side2}} - X_{R_{side1},C_{side1}}\right)^2 + \left(Y_{R_{side2},C_{side2}} - Y_{R_{side1},C_{side1}}\right)^2} \tag{8}$$

where:

$$\begin{aligned} R_{side1} &= R_{side2} = R\_ROI_{\min} + \frac{100\ mm}{l_{ROI}} \\ C_{side1} &= \min(C_{ROI})\ \forall\ SR_{R_{side1},C\_ROI} \in Pi \\ C_{side2} &= \min(C_{ROI})\ \forall\ SR_{R_{side2},C\_ROI} \in Pi \end{aligned} \tag{9}$$

If the first criterion is not met, the object is discarded as a person, it is eliminated from *D* and the second criterion will not be taken into account.

The second criterion is based on the fact that the contour of the upper part of the head (*h_contour*) projected in the plane (X, Y) has a parabolic shape, see Figure 7. To verify that this criterion is fulfilled, with all mean values of the coordinates (X, Y) of the subregions that make up the perimeter of the head, the curve of the parabola closest to that perimeter is determined. We use the Levenberg-Marquardt algorithm to find the least-squares set of coefficients ($coef1, coef2$) that best fit the set of input data points (X, Y) to a parabolic function *f(X)*. Figure 8 shows the set of points of the contour head (red) and the values obtained with the fitted curve (blue).

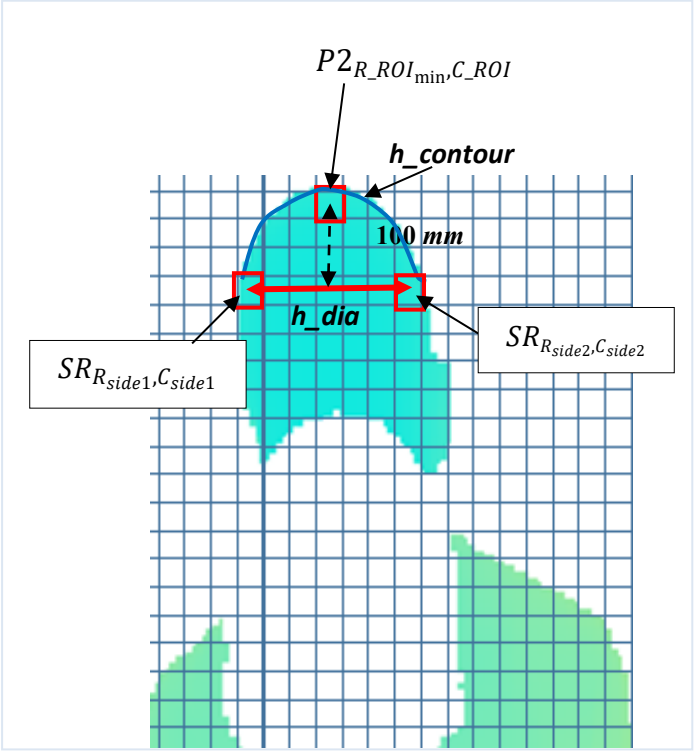


Figure 7. Graphic representation of the geometric parameters used

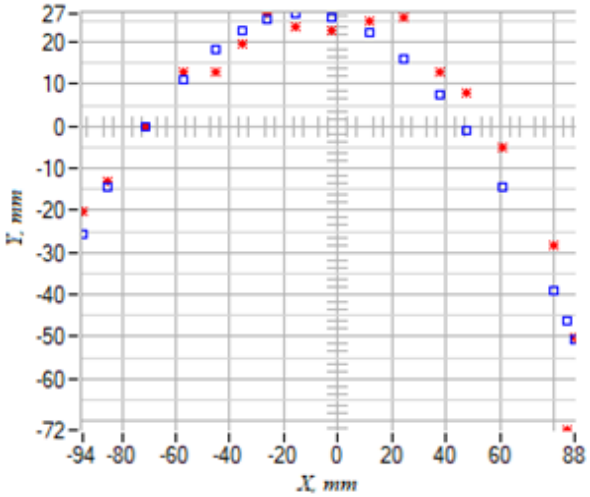


Figure 8. Contour head (red) and fitted parabola (blue).

The coordinates (X_w, Y_w) of the subregion w of the head are: $X_w = X_{R_w,C_w}$ and $Y_w = Y_{R_w,C_w}$, being $w = 1, 2, 3, \cdots, (C_{side2} - C_{side1})$; $C_w = C_{side1} + w - 1$ and $R_w = RI\_ROI_{C_w}$

We used the function for each subregion w of the border (eq. (9)):

$$Y_w = coef1 \cdot (X_w - coef2)^2 + Y_{max} \tag{9}$$

where, $Y_{max} = \max(Y_w)$.

To classify objects as people or not people, we take into account the following three parameters θ_i $(i = 1, .., 3)$:

- θ_1 is the mean squared error generated by the difference between the fitted curve and the input data points (X, Y).
- θ_2 is Coordinate Y of the parabola focus: $\theta_2 = Y_{max} - 0.25/coef1$
- θ_3 is Coordinate X of the parabola vertex: $\theta_3 = coef2$

Each parameter θ_i can determine if an object is a person, taking into account a certain threshold Th_i. If the following condition is fulfilled for all three parameters, the object is classified as a person.

$$\left\| \theta_i - \widehat{\theta_i} \right\| < Th_i \tag{10}$$

The $\widehat{\theta_i}$ and Th_i values were determined experimentally using set of people with different characteristics, and in different scene positions. The obtained values are shown in Table 1.

Table 1. Value and threshold for parameters $\boldsymbol{\theta_i}$

Parameter	$\widehat{\theta_i}$ (mm)	Th_i (mm)
θ_1	0	44
θ_2	76	52
θ_3	0	41

The $\widehat{\theta_i}$ is the average value of the θ_i set and $Th_i = 2.58 \cdot \sigma_i$, were σ_i is the standard deviation of the θ_i set. These values are chosen to try to achieve a 99% confidence interval.

In order to obtain the Th_i values, an analysis of the behavior of each of them was performed based on the distance between the head and the camera d_{cam_head} , in our case the used camera is an Intel Realsense D435. From this analysis, it was found that from 3m away the standard deviation of the parameters increases abruptly. Therefore, from 3m away the errors in the detection of people will increase significantly. Figure 9 shows as an example the standard deviation $\sigma_1$of parameter θ_1 for distances between 1m and 4.2m.

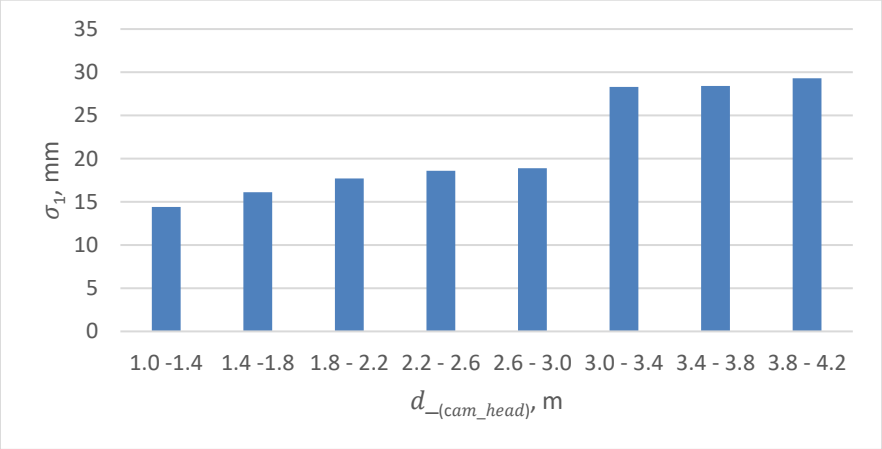


Figure 9.Standard deviation $\boldsymbol{\sigma_1}$ vs distance between the camera and the head $\boldsymbol{d_{cam\_head}}$.

3. Results and discussion

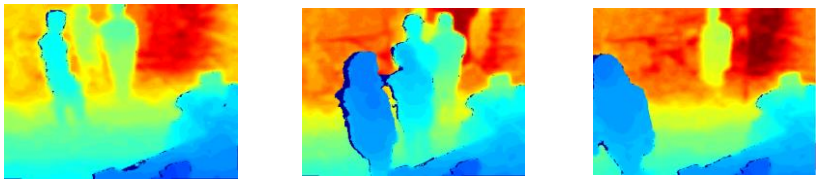
3.1 Experimental setup

3.1.1. Datasets

In order to validate the operation of the proposed method, several experiments have been carried out, using three public datasets, captured by three sensors with different

measurement principles and spatial resolutions. These datasets are briefly described below. It is worth highlighting that the differences between the three datasets (type of depth sensor, camera height and angle, image resolution, etc.) allow testing the robustness of the proposal against these parameters.

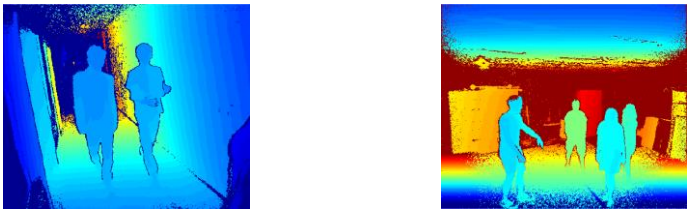
- GFPD dataset. (Fuentes-Jimenez et al. 2020b) is a dataset recorded with a high resolution (1280 x 720) Intel® RealSense™ D435® sensor. The measurement principle of this sensor is active stereo. The recording covers a broad variety of conditions: the sensor was located at different heights (h_{sensor} from 2200 mm to 2720 mm) with tilt angle α from 26 to 41 degrees, the sequences were captured in scenarios with different background and different natural (solar) and artificial lighting, as it can be seen in the examples showed in Figure 10. All depth images are represented using a colormap that represents the different depth values using different colors.



(a) (b) (c)

Figure 10. Examples of images of the GFPD datasets. a) Scenario 1 with sunlight entering through the windows. b) Scenario 2 at night. c) Scenario 2, where there are no windows in front of the camera.

- EPFL datasets (Bagautdinov et al., 2015). It is a dataset recorded with a Kinect® v2 sensor (512 x 424 pixels). The measurement principle of this sensor is ToF. The sequences were captured in two different scenarios: a corridor in a university building with up to 8 people and a laboratory with up to 4 people. Examples of both scenarios are shown in Figure 11.



(a) (b)

Figure 11. Examples of images of the EPFL datasets. a) EPFL corridor. b) EPFL Lab.

- KTP dataset (Munaro & Menegatti, 2014): It is a dataset recorded with a Kinect® v1 sensor (640 x 480). The measurement principle of this sensor is structured light. This dataset contains several sequences with up to 5 people walking in a laboratory. We only use sequences captured in a static position (KTP Still). An image showing the scenario used in this dataset is shown in Figure 12.

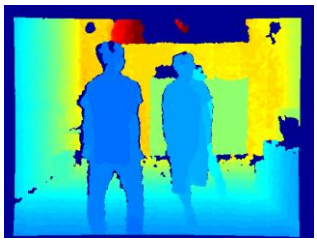


Figure 12. Example of image of the KTP dataset.

3.1.2 Hardware and software implementation.

Since the fundamental motivation of this work is to look for a method that allows people to be detected in real time using low-cost hardware, we have carried out the tests on two different computers whose characteristics are detailed below.

- **Computer 1**. Laptop; CPU: Intel® i7-6500u, 2.5 GHz, 8GB of RAM; OS: Windows 10.
- **Computer 2**. Desktop; CPU: Intel® Core™2 Quad Q9550 2.83 GHz, 4GB of RAM; OS: Windows 8.

The entire implementation has been developed in the standard C programming language, without using libraries such as OpenCV. The use of standard C allows to optimize the developed algorithm, reducing the computation time. In addition, it eases the portability between different platforms, without the need to install libraries.

3.2 Performance evaluation

This section shows the results of the detection of people without the use of any tracking algorithm, for distances less than three meters and only considering people who do not have partially or totally occluded their heads. The values of the biometric parameters used in the classification process for the three databases are those determined in section 2.3.

Table 2 shows the results obtained for each of the databases, indicating the total number of people (#People), the number of false positives (FP) and false negatives (FN), precision, recall, and F1-score.

Table 2. Experimental results obtained with the three different datasets using our proposal.

Dataset	#People	FP	FN	Precision	Recall	F1-score
GFPD	7970	18	91	99.77%	98.86%	99.31±0.07%
KTP	1637	29	101	98. 15%	93.83%	95.94±0.26%
EPFL Lab.	820	7	151	98.96%	81.59%	89.44±0.01%
EPFL Corr.	3534	15	132	99.56%	96.26%	97.88±0.31%

The precision is similar in all datasets, being greater to 98%. It is worth noting the high precision obtained in EPFL-corridor, where there is a high degree of occlusion due to the characteristics of the dataset.

After analyzing the errors, it can be observed that the number of false positives is mainly due to body parts (usually the shoulders) of people who have their head occluded or out of the scene.

The number of false negatives depends on several causes. A common cause in the different datasets is due to the method itself and is caused by not processing a person when he is in front of another at a distance less than $d_{min\_betw\_obj}$. This error can be eliminated using a more efficient edge detection algorithm, but with a significant increase in the computational cost.

Another cause of the occurrence of false negatives is due to errors in the depth measurements provided by the cameras, caused by their operating principle. In the case of camera Intel Realsense D435, deformations of the head occur, mainly at the lateral edges of the scene. This may be due to the perspective distortion, the reflectivity of the head, the power of the projector, etc. In general, false negatives do not occur in consecutive images, so the number of these can be reduced using tracking techniques.

In the case of measurements obtained with camera Kinect v1 and Kinect v2, the main errors are because the reflectivity of the object has a significant influence. In some people because of their hair characteristics, the light is not reflected and therefore there are no 3D measurements of the area covered with hair. In Figure 11 b) it can be seen how the person who is on the left side of the image has a cropped head, however other people who are further away from the camera have very few distortions. In EPFL lab most of the false negatives correspond to this person in several consecutive frames, regardless of the distance from the camera. Figure 12 shows that the person closest to the camera is missing part of the head in the depth image. The missing measurements are represented in black color in the images, because the camera returns a value 0 in these pixels.

The results obtained with our method regarding evaluation metrics (precision, recall, and F1-score) have been compared against the method proposed by (Redmon & Farhadi, 2018) (YOLO v3) for being this one of the most used in the state-of-the-art methods based on Convolutional Neural Networks (CNN) (Zhao et al., 2019). Table 3 shows the metrics of using RGB images (YOLO v3 RGB), YOLO v3 modified to use depth images, and trained using the GFPD dataset (YOLO v3 Depth) and our method. In the first case, YOLO v3 RGB was trained using COCO dataset (Lin et al. 2014) with the object detection modality, which contains more than 200.000 RGB images and 80

objects categories, which allow the network to differentiate correctly between the detected persons and a big variety of different objects, making the network more robust to human-similar objects. In the second case, YOLO v3 Depth feature extractor was first pretrained using an autoencoder configuration with GOTPD dataset (Fuentes-Jiménez et al. 2020), which is an overhead depth images dataset with 51418 frames, The first pretrain helps the feature extractor to learn the typical structure of depth images with people, After this first pretrain we train the entire detector with GESDPD database (Martín-López et al. 2020), which is a synthetic depth images dataset for the person detection task. Once we finalize the two previous process, we finally fine-tune the detector with a smaller dataset like GFPD (Fuentes-Jiménez et al. 2020b). These processes are necessary to ensure the best training of YOLO v3 Depth. To further specify the use of these non-yolo-specific databases, GESDPD was labelled automatically, while GFPD was manually labelled in YOLO bounding box format. For this, the three databases have been used, considering people located at less than 3 meters. In the case of the GFPD dataset, the comparison has been carried out using only those sequences where RGB and depth images were available, that in the future we will denominate as GFPD reduced.

As it can be observed in Table 3, the best results are obtained for the GFPD (reduced) and EPFL corridor datasets, for which our proposal outperforms the other approaches. There are especially noteworthy the results obtained for EPFL-corridor, where there is a high degree of occlusions that cause errors in the other methods, as it can be seen in (Bagautdinov et al., 2015). In the case of EPFL-Lab, our method obtains the highest precision, but the values of recall and F1-score are slightly worse than for YOLO v3 RGB approaches. This is due to the number of false negatives caused by measurement errors that lead to distortions in the shape of the person's head (as can be seen in the example in Figure 11(b), where the head of the person on the left is incomplete), that does not affect to the RGB images used by YOLO v3 RGB. Regarding the results with KTP dataset, there are slightly better for YOLO v3 RGB, it is again due to the erroneous depth measures in the dataset, which affect the shape of people's heads (as it can be seen in the examples shown in Figure 11 and Figure 12).

In the case of YOLO v3 depth, the results are significantly worse for all datasets. This may be due to errors in depth measurements (as shown in figures 11 and 12), along with the lack of large depth datasets for training the network.

Additionally, we have provided F1-score confidence bands for each of the methods and datasets used, to ensure the statistical significance. As it can see the confidence bands of Our proposal and YOLO v3 RGB are quite small, which indicates a big certainly on the values provided while the values of YOLO v3 Depth are bigger compared with the previous two methods, introducing greater uncertainty into its results. As we can see none of the results reported with their confidence bands overlap between them, which leads us to establish a statistical significance on the reported results.

Table 3. Results of the comparison of the results obtained with the proposed method and YOLO v3, CNN-based methods.

Dataset	Our proposal			YOLO v3 RGB			YOLO v3 Depth		
	Prec.	Recall	F1-sc.	Prec.	Recall	F1-sc.	Prec.	Recall	F1-sc.
GFPD reduced	**99.78**	**97.48**	**98.62 ±0.07**	86,72	89,46	88,07 ±1.13	84,92	66,21	74,41 ±2.78
KTP	98.15	93.83	95.94 ±0.26	**99.32**	**98.71**	**99.01 ±0.34**	92,46	75,23	82,96 ±1.45
EPFL Lab.	**98.96**	81.59	89.44 ±0.01	95.65	**99.29**	**97.44 ±0.21**	91,32	92,25	91,78 ±0.87
EPFL Corr.	**99.56**	**96.26**	**97.88 ±0.31**	74.36	69.42	71.81 ±1.72	82,27	53,42	64,78 ±2.43

3.3 Computational demands.

To analyze the computational efficiency of the proposed algorithm, we have determined the average times of each step. We have chosen sequences with more than three people for a fair comparison, because in our proposal there is a direct relationship between the number of people and the computation time. Table 4 shows the results for the two computers and for the three sensors with different resolutions used. In steps 1 to 3, the number of people present in the scene has very little influence on the processing time. This time depends directly on the resolution of the sensor and is approximately

90% of the total time. Steps 4 to 7 are repeated for each object detected. Therefore, its processing time will depend on the number of people present in the scene.

From the above, it can be deduced that the ratio of the total processing time to the resolution of the sensor is practically constant for a given computer, for Computer 1, it is approximately 4 ms/megapixel and for Computer 2 it is approximately 17 ms/megapixel.

Table 4. Average computational cost for each algorithm step in both used computers.

CPU	**Computer 1**			**Computer 2**		
Sensor	D435	Kinect v1	Kinect v2	D435	Kinect v1	Kinect v2
Spatial resolution. pixels	921600	307200	217088	921600	307200	217088
Average time of the algorithm steps (ms)						
1. Remove background	*2.41*	*0.79*	*0.58*	*9.01*	*2.90*	*2.52*
2. Divide D in subregions (SR)	*0.91*	*0.31*	*0.21*	*5.39*	*1.62*	*1.19*
3. Find the objects in each column	*0.01*	*0.003*	*0.002*	*0.05*	*0.03*	*0.01*
4. Find object maximum *5. Find ROI around the maximum* *6. Process the object in the ROI* *7. Biometric classification (> 3 people)*	*0.42*	*0.21*	*0.18*	*0.98*	*0.65*	*0.32*
Total (ms)	**3.75**	**1.323**	**0.972**	**15.43**	**5.20**	**3.94**

If the processing times of recent works are observed, regardless of the sensor used and its position with respect to the scene, results are worse than ours. In (Sun et al., 2019) for 320 x 240 zenithal RGB-D images there are processing times of 16.5 ms (1.1 ms of background removal and 15.4 ms of head identification) in a CPU. Regarding YOLO v3 RGB and YOLO v3 depth, both approaches require a GPU (it has been used a GTX 1080), obtaining a processing time of 47.62ms (21 fps) for RGB and 36.3 ms (27.3 fps) for depth, being these times practically independent of the image resolutions.

4. Conclusions

This paper shows a robust and fast method of detecting people using depth images acquired with a camera located in a frontal position. In the design of each of the steps of the method, it has been considered to reduce the computational cost and due to the simplicity of the method, in its implementation, only the standard C libraries have been used.

In this work we have compared the results obtained with our proposal with YOLO v3 because this is a comparative reference in the most recent detection and classification works. This comparison has been made for YOLO v3 using RGB and depth images and our proposal that only uses deep images. In the case of RGB images YOLO v3 has better results in the KTP and EPFL Lab datasets, because in some of the depth images, the heads of some people appear deformed or incomplete, due to the reflectivity of people hair. However, in the GFPD and EPFL Corridor datasets, we surpass the results of YOLO v3, although there are complex datasets, with multiple people and a high degree of occlusions.

In the case of using only depth image, our proposal exceeds YOLO v3 in the three used datasets. These better results are because in the images of depth, captured with the camera based on stereo vision, the distortions of the heads are much smaller than with the cameras Kinect v1 and Kinect V2. In the case of EPFL Corridor, in this scenario the distortions of the heads are less than EPFL Lab.

The main contribution of our method is relative to the computational cost, where our proposal far exceeds the works found in the current state of the art. We have carried out a study of the calculation times of our algorithm. For this, we have evaluated the average processing times of each step in two different CPUs, using the images of the three databases (three different resolutions). The results shown in table 4 (15.43 of total processing time of the high-resolution images, using an old dual core CPU) validate that the system can operate in real time on very low-cost processors.

In future work, we will intend to improve the edge detection algorithm used in point 2 of Section 2.2, with the aim of improving the detection results when two people are very close to each other.

Acknowledgements

This work has been supported by the Spanish Ministry of Economy and Competitiveness under project HEIMDAL-UAH (TIN2016-75982-C2-1-R) and by the University of Alcalá under projects ACERCA (CCG2018/EXP-029) and ACUFANO (CCG19/IA-024).

References

Bagautdinov, T., Fleuret, F., & Fua, P. (2015). Probability occupancy maps for occluded depth images. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *07-12-June-2015*, 2829–2837. https://doi.org/10.1109/CVPR.2015.7298900

Bevilacqua, A., Di Stefano, L., & Azzari, P. (2006). People Tracking Using a Time-of-Flight Depth Sensor. *Video and Signal Based Surveillance, 2006. AVSS '06. IEEE International Conference On*, 89. https://doi.org/10.1109/AVSS.2006.92

Beyl, T., Nicolai, P., Raczkowsky, J., Worn, H., Comparetti, M. D., & De Momi, E. (2013). Multi kinect people detection for intuitive and safe human robot cooperation in the operating room. *2013 16th International Conference on Advanced Robotics (ICAR)*, 1–6. https://doi.org/10.1109/ICAR.2013.6766594

Da Silva Guizi, F., & Kurashima, C. S. (2016). Real-time people detection and tracking using 3D depth estimation. *Proceedings of the International Symposium on Consumer Electronics, ISCE*, 39–40. https://doi.org/10.1109/ISCE.2016.7797359

Del Pizzo, L., Foggia, P., Greco, A., Percannella, G., & Vento, M. (2016). Counting people by RGB or depth overhead cameras. *Pattern Recognition Letters*, *81*, 41–50. https://doi.org/10.1016/j.patrec.2016.05.033

Dumoulin, J., Canevet, O., Villamizar, M., Nunes, H., Khaled, O. A., Mugellini, E., Moscheni, F., & Odobez, J.-M. (2018). UNICITY: A depth maps database for people detection in security airlocks. *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–6. https://doi.org/10.1109/AVSS.2018.8639152

Fang, Y., Xie, J., Dai, G., Wang, M., Zhu, F., Xu, T., & Wong, E. (2015). 3D deep shape descriptor. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *07-12-June*, 2319–2328. https://doi.org/10.1109/CVPR.2015.7298845

Fernandez-Rincon, A., Fuentes-Jimenez, D., Losada-Gutierrez, C., Marron-Romera, M., Luna, C. A., Macias-Guarasa, J., & Mazo, M. (2017). Robust People Detection and Tracking from an Overhead Time-of-Flight Camera. *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, *Visigrapp*, 556–564. https://doi.org/10.5220/0006169905560564

Fuentes-Jimenez, D., Martin-Lopez, R., Losada-Gutierrez, C., Casillas-Perez, D., Macias-Guarasa, J., Luna, C. A., & Pizarro, D. (2020). DPDnet: A robust people detector using deep learning with an overhead depth camera. *Expert Systems with Applications*, *146*. https://doi.org/10.1016/j.eswa.2019.113168

Fuentes-Jimenez, D; Losada-Gutierrez, C.; Macias-Guarasa, J; Luna, C. & Pizarro, D. (2020). Depth Person detection database (GFPD-UAH)}, *Kaggle*, DOI=10.34740/KAGGLE/DSV/915669, https://www.kaggle.com/dsv/915669.

Ghiţă, Ş. A., Barbu, M. Ş., Gavril, A., Trăscău, M., Sorici, A., & Florea, A. M. (2018). User detection, tracking and recognition in robot assistive care scenarios. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10965 LNAI*, 271–283. https://doi.org/10.1007/978-3-319-96728-8_23

Jeong, C. Y., Choi, S., & Han, S. W. (2013). A method for counting moving and stationary people by interest point classification. *Image Processing (ICIP), 2013 20th IEEE International Conference On*, 4545–4548. https://doi.org/10.1109/ICIP.2013.6738936

Jia, L., & Radke, R. J. (2014). Using Time-of-Flight Measurements for Privacy-Preserving Tracking in a Smart Room. *IEEE Transations on Industrial Informatics*, *10*, 689–696.

Jimenez, D., Pizarro, D., Mazo, M., & Palazuelos, S. (2012). Modelling and correction of multipath interference in time of flight cameras. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 893–900. https://doi.org/10.1109/CVPR.2012.6247763

Khan, M. H., Shirahama, K., Farid, M. S., & Grzegorzek, M. (2017, January 10). Multiple human detection in depth images. *2016 IEEE 18th International Workshop on Multimedia Signal Processing, MMSP 2016*. https://doi.org/10.1109/MMSP.2016.7813385

Lee, S., Kang, B., Kim, J. D. K., & Kim, C. Y. (2012). Motion blur-free time-of-flight range sensor. *IS&T/SPIE Electronic Imaging*, *8298*. http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1284120

Liciotti, D., Paolanti, M., Frontoni, E., & Zingaretti, P. (2017). People Detection and Tracking from an RGB-D Camera in Top-View Configuration: Review of Challenges and Applications. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10590 LNCS*, 207–218. https://doi.org/10.1007/978-3-319-70742-6_20

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer, Cham.

Luna, C. A., Losada, C., Fuentes-Jimenez, D., Fernandez-Rincon, A., Mazo, M., & Macias-Guarasa, J. (2016). Robust People Detection Using Depth Information from an Overhead Time-of-Flight Camera. *Expert Systems with Applications*, *71*, 240–256. https://doi.org/10.1016/j.eswa.2016.11.019

Martín-López, R.; Fuentes-Jiménez, D.; Luengo-Sánchez, S.; Losada-Gutiérrez, C.; Marrón-Romera, M. and Luna, C. (2020). Towards Deep People Detection using CNNs Trained on Synthetic Images.In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications* - Volume 5: VISAPP, ISBN 978-989-758-402-2, pages 225-232. DOI: 10.5220/0008879102250232

Migniot, C., & Ababsa, F. (2013). 3D human tracking in a top view using depth information recorded by the Xtion Pro-Live camera. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *8034 LNCS*(PART 2), 603–612. https://doi.org/10.1007/978-3-642-41939-3_59

Munaro, M., & Menegatti, E. (2014). Fast RGB-D people tracking for service robots. *Autonomous Robots*, *37*(3), 227–242. https://doi.org/10.1007/s10514-014-9385-0

Muñoz-Salinas, R., Aguirre, E., & García-Silvente, M. (2007). People detection and tracking using stereo vision and color. *Image and Vision Computing*, *25*(6), 995–1007. https://doi.org/10.1016/j.imavis.2006.07.012

Muñoz-Salinas, R., Aguirre, E., García-Silvente, M., & Gonzalez, A. (2005). *People Detection and Tracking Through Stereo Vision for Human-Robot Interaction* (pp. 337–346). https://doi.org/10.1007/11579427_34

Pereira, F. G., Vassallo, R. F., & Salles, E. O. T. (2013). Human–Robot Interaction and Cooperation Through People Detection and Gesture Recognition. *Journal of Control, Automation and Electrical Systems*, *24*(3), 187–198. https://doi.org/10.1007/s40313-013-0040-3

Ramanan, D., Forsyth, D. A., & Zisserman, A. (2006). Tracking People by Learning Their Appearance. *Pattern Analysis and Machine Intelligence, IEEE Transactions On*, *29*(1), 65–81. https://doi.org/10.1109/tpami.2007.250600

Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. http://arxiv.org/abs/1804.02767

Sell, J., & O'Connor, P. (2014). The Xbox One System on a Chip and Kinect Sensor. *IEEE Micro*, *34*(2), 44–53. https://doi.org/10.1109/MM.2014.9

Sidla, O., Lypetskyy, Y., Brändle, N., & Seer, S. (2006). Pedestrian detection and tracking for counting applications in crowded situations. *Proceedings - IEEE International Conference on Video and Signal Based Surveillance 2006, AVSS 2006*. https://doi.org/10.1109/AVSS.2006.91

Smisek, J., Jancosek, M., & Pajdla, T. (2011). 3D with Kinect. *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference On*, 1154–1160. https://doi.org/10.1109/ICCVW.2011.6130380

Solbach, M. D., & Tsotsos, J. K. (2017). Vision-Based Fallen Person Detection for the Elderly. *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, *2018-January*, 1433–1442. https://doi.org/10.1109/ICCVW.2017.170

Stahlschmidt, C., Gavriilidis, A., Velten, J., & Kummert, A. (2013). People Detection and Tracking from a Top-View Position Using a Time-of-Flight Camera. *Communications in Computer and Information Science*, *368 CCIS*, 213–223. https://doi.org/10.1007/978-3-642-38559-9_19

Sumalan, A. L., Ichim, L., & Popescu, D. (2018). Person Detection in Video Surveillance. *2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 1–6. https://doi.org/10.1109/ECAI.2018.8678939

Sun, S., Akhtar, N., Song, H., Zhang, C., Li, J., & Mian, A. (2019). Benchmark Data and Method for Real-Time People Counting in Cluttered Scenes Using Depth Sensors. *IEEE Transactions on Intelligent Transportation Systems*, *20*(10), 3599–3612. https://doi.org/10.1109/tits.2019.2911128

Tian, L., Li, M., Hao, Y., Liu, J., Zhang, G., & Chen, Y. Q. (2018). Robust 3-D human detection in complex environments with a depth camera. *IEEE Transactions on Multimedia*, *20*(9), 2249–2261. https://doi.org/10.1109/TMM.2018.2803526

Tomoya, A., Nakayama, S., Hoshina, A., & Sugaya, M. (2017). A Mobile Robot for Following, Watching and Detecting Falls for Elderly Care. *Procedia Computer Science*, *112*, 1994–2003. https://doi.org/10.1016/j.procs.2017.08.125

Wang, C., Ma, X., & Liao, C. (2018). Real-time people detection from a top-view ToF camera. In X. Jiang & J.-N. Hwang (Eds.), *Tenth International Conference on Digital Image Processing (ICDIP 2018)* (p. 288). SPIE. https://doi.org/10.1117/12.2503254

Wojek, C., & Schiele, B. (2008). A performance evaluation of single and multi-feature people detection. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *5096 LNCS*, 82–91. https://doi.org/10.1007/978-3-540-69321-5_9

Yang, C.-J., Chou, T., Chang, F.-A., Ssu-Yuan, C., & Guo, J.-I. (2016). A smart surveillance system with multiple people detection, tracking, and behavior analysis. *2016 International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*, 1–4. https://doi.org/10.1109/VLSI-DAT.2016.7482569

Yimyam, W., Kocento, K., & Ketcham, M. (2018). Video Surveillance System Using IP Camera for Target Person Detection. *2018 18th International Symposium on Communications and Information Technologies (ISCIT)*, 176–179. https://doi.org/10.1109/ISCIT.2018.8587927

Yoganandan, N., Pintar, F. A., Zhang, J., & Baisden, J. L. (2009). Physical properties of the human head: Mass, center of gravity and moment of inertia. *Journal of Biomechanics*, *42*(9), 1177–1192. https://doi.org/10.1016/j.jbiomech.2009.03.029

Zhang, X., Yan, J., Feng, S., Lei, Z., Yi, D., & Li, S. Z. (2012). Water Filling: Unsupervised People Counting via Vertical {K}inect Sensor. *Advanced Video and Signal-Based Surveillance (AVSS), 2012 IEEE Ninth International Conference On*, 215–220. https://doi.org/10.1109/AVSS.2012.82

Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object Detection with Deep Learning: A Review. In *IEEE Transactions on Neural Networks and Learning Systems* (Vol. 30, Issue 11, pp. 3212–3232). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/TNNLS.2018.2876865