Feature extraction for real-time human gait recognition using DensePose and KeyPoints by Detectron2

Colaboración
José Misael Burruel Zazueta; Héctor Rodríguez Rangel, Tecnológico Nacional de México Campus Culiacán; Luis Alberto Morales Rosales; Conacyt-Universidad Michoacana de San Nicolás de Hidalgo Vicenc Puig Cayuela; Gloria Ekaterine Peralta Peñuñuri, Institut de Robòtica i Informática Industrial, Universitat Politécnica de Catalunya, Consejo Superior de Investigaciones Científicas, Parc Tecnológic de Barcelona

Fecha de recepción: 17 de enero del 2023
Fecha de aceptación: 08 de noviembre del 2023

ABSTRACT: Human Gait Recognition (HGR) is a technique that aims to identify people by their gait. A process of utmost importance for HGR is extracting features necessary for identification. This process takes most of the recognition time, so developing a methodology capable of performing feature extraction in the shortest amount of time is necessary. This paper describes a methodology to extract human body features from two approaches (key points and silhouettes) in less than 50 milliseconds per frame. According to the proposal, extracting the features of a complete walking sequence (2 to 3 seconds of video) in less than 30 seconds is possible. Low time allows the feature extraction process to be performed in real-time, with a half-minute response time.

KEY WORDS: HGR, silhouettes, key points, FPS, gait, features, angles.

RESUMEN: El reconocimiento del andar humano (RAH) es una técnica cuyo objetivo es identificar a las personas por su forma de caminar. Un proceso de suma importancia para el RAH es la extracción de las características necesarias para la identificación. Este proceso ocupa la mayor parte del tiempo de reconocimiento, por lo que es necesario desarrollar una metodología capaz de realizar la extracción de características en el menor tiempo posible. Este trabajo describe una metodología para extraer características del cuerpo humano a partir de dos aproximaciones (puntos clave y siluetas) en menos de 50 milisegundos por fotograma. Según la propuesta, es posible extraer las características de una secuencia completa de caminata (de 2 a 3 segundos de vídeo) en menos de 30 segundos. El bajo tiempo permite realizar el proceso de extracción de características en tiempo real, con un tiempo de respuesta de medio minuto.

PALABRAS CLAVE: HGR, siluetas, puntos clave, FPS, marcha, rasgos, ángulos.

INTRODUCTION

Global insecurity has remained at high levels in recent years [1]. Some crimes, such as assaults, home robberies, kidnappings, and murders, are the most alarming to the general population due to the seriousness and frequency with which they occur. The governments of the countries most affected by these types of crimes have implemented different security mechanisms to prevent the inhabitants or to identify suspicious people's behavior. Surveillance through security cameras placed at strategic points in cities is one of the most common ways. One way to enhance these surveillance mechanisms is to implement people identification systems that take advantage of the images captured by security cameras.

People identification is fundamental in many applications. The recent increase in global crime has led to an increase in efforts to investigate new techniques for people identification along with the artificial intelligence advance. Some of these techniques are based on biometric people features, i.e., they identify people through human physical characteristics or behaviors.

Several biometric features are currently used in security systems to recognize a person, e.g., fingerprint [2], face [3], voice [4], eye iris [5], and veins [6], among others. The main idea of using this type of feature for a security system is to guarantee a correct recognition of the person's identity possessing such features.

During the last decade, deep learning has produced interest in research groups with promising and outstanding results in many areas, such as natural language processing (NLP), texture classification, object recognition, face recognition, speech recognition, information retrieval, and traffic sign classification [7]. These results mean deep learning techniques are suitable for performing image or time series processing tasks to classify behaviors or objects based on their main characteristics. For these reasons, human beings' recognition through biometric features is not alien to deep learning.

One way to implement biometric recognition techniques in video surveillance is Human Gait Recognition (HGR). HGR is a biometric technique responsible for identifying a person through how he/she walks. In the 21st century, some researchers have worked on this biometric feature because it offers a primary advantage, running silent recognition at a distance and even without the consent or knowledge of the walking subject [8]. Although this technique is relatively new, it has been an active research area, especially in the last decade. Several articles guarantee a 90% successful recognition rate, comparable to the success rates of the most widely used biometric systems.

The researchers most engaged in HGR divide this recognition technique (image-based) into two different approaches. The model-based approach [9] mainly performs features extraction specific to the subject's behavior, i.e., flexion angles of limbs, torso, head, hip and their possible combinations (e.g., head tilt concerning the torso), stride, stride cadence, among others. The second approach is based on extracting the complete subject silhouettes from the images and performing different image processing and recognition strategies based on these silhouettes. This approach is based on appearances [9], and several methodologies are used for HGR. The most employed option within the different research in the state-of-the-art is the recognition through energy images (GEI). A GEI is a spectrum representation that gathers all silhouettes into a single image for feature extraction and subsequent classification [10].

The main advantage of the model-based approach is its robustness to occlusions that may partially cover the person's body being analyzed. This approach's main disadvantage is its high computational power consumption, which can cause delays in information processing. On the other hand, the main advantage of the appearance-based approach is its low computational power cost. At the same time, its disadvantage is its low resistance to occlusions present in the images. Consequently, both approaches advantages and disadvantages are counterbalanced, offering different qualities depending on the desired application scenario.

It is due to the above that it is necessary to develop more and new algorithms that allow the features' extraction of people's gait. Generating new options will allow developers and researchers to have a range of opportunities according to the development characteristics they are looking for. Very few works in the field of research express the time it takes to perform each of the processes because these times may vary depending on the operating conditions of the computer equipment.

Contributions

The contributions of this work are described as follows:
1. The use of the DensePose and KeyPoints neural networks developed by Facebook at Detectron2, which are based on ResNet50 for feature extraction based on both approaches, is proposed.
2. A hardware configuration is proposed for the processing and implementing of the described neural networks that allow feature extraction in less than 50 milliseconds per video frame.
3. A repository provides the ten videos used during the experimentation, the source code used for feature extraction, and the images and time series that resulted from the process.

The ability to process and extract features from a video in less than 50 milliseconds per frame enables real-time HGR. This fast extraction time leads to a scenario where under normal video parameters (HD, 20fps) there would be a delay of one second for every second of video in the walk cycle. Thus, assuming that the walk cycle typically takes two to three seconds, the extraction process takes approximately the same amount of time. In summary, the above contributions provide a time advantage and improve the quality of the extracted features for further processing, especially in HGR applications

MATERIALS AND METHODS

Extracting features directly from the images extracted in each video frame is a standard process within computer vision. Describing the different hardware materials, providing the videos used for this work development, and the source code aims to allow anyone to replicate the methodology specified below.

Materials

The system was developed in Python code, and the experiments were performed on three different computers. The first computer used was an iMac with a Quad Core Intel i5 processor with 16 GB RAM and 2.9 GHz speed. The second computer used has a Quad Core Intel i5 processor with 32 GB RAM and 3.4 GHz speed. Finally, the third computer has the same processor characteristics as the second one. However, two NVIDIA GeForce GTX 1080 GPUs were added.

Ten videos of test subjects walking down an 8-meter-long corridor were used. The camera was located 5 meters away from the corridor and at a 90-degree angle to the corridor. Each video was approximately 2 to 4 seconds long (depending on walking speed) at 30 frames per second and a 640x480 pixels size. Figure 1 shows the graphical presentation of the scenario in which the video sequences used in the experiment were taken.

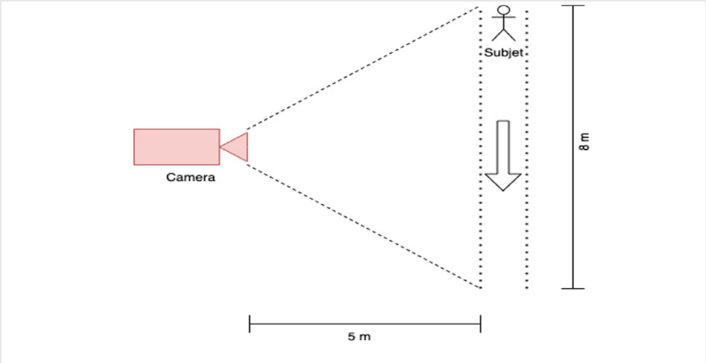

Figure 1. Scenario for taking video sequences.
Source: own elaboration.

Two pre-trained convolutional neural networks (CNNs) available in Detectron2 [11] were used to process each video. The first CNN is called DensePose [12], which detects and extracts the total or partial silhouettes of the subject identified in the image. The second CNN, KeyPoints [11], is used to identify the human body key points, i.e., it locates the different coordinate points that connect the human body in the image. In total, 15 key points directly impact the HGR: ankles, knees, hips (left, right, and midpoint), shoulders (left, right, and midpoint), elbows, wrists, and head. However, the CNN KeyPoints also provide the key points corresponding to the eyes and ears.

Both CNNs are based on an architecture widely used in deep learning called ResNet50 [13].

Methodology

The developed feature extraction system divides the videos frame by frame. These frames are sent to each neural network to extract silhouettes (DensePose) and key points (KeyPoints). Figure 2 shows the image data flow. The frames are sent directly to the CNNs DensePose and KeyPoints.

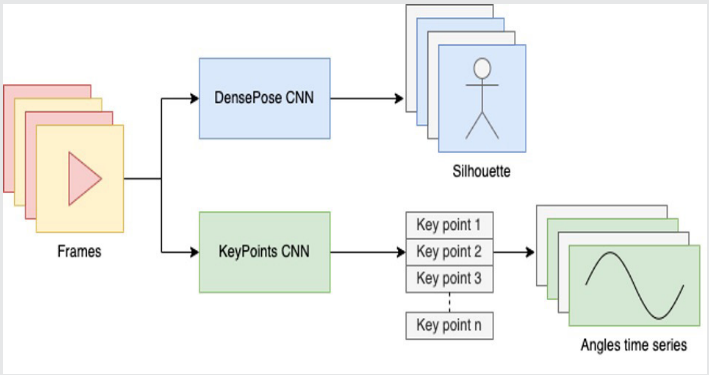

Figure 2. Video feature extraction process.
Source: own elaboration.

DensePose detects and segments the subjects' silhouettes to erase the image background. Then, the silhouettes are stored to generate a new video with the segmented silhouettes. Figure 3 shows the extracted silhouette in a video frame.

KeyPoints, on the other hand, detects the previously defined key points in vector forms of three variables for each key point. Two of these three variables correspond to the location coordinates on the horizontal and vertical axes of the key point in question. The third variable defines the probability that the key point is or is not in the position defined by the first two variables. For the experimentation purposes in this work, the third variable is not considered because a low key points probability would produce information gaps in the vectors, i.e., there would be variations in the time series magnitude, and the intention is to obtain same-size vectors.

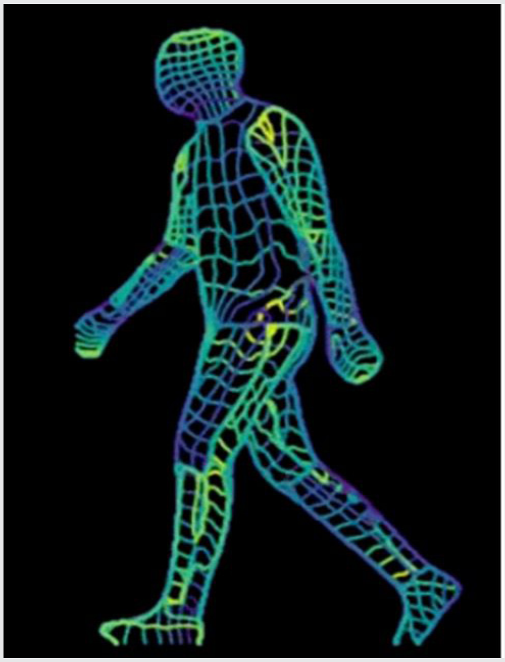

Figure 3. Silhouette extracted from a video frame by DensePose CNN.
Source: own elaboration.

Because KeyPoints CNN provides only the key points mentioned above as output, it is necessary to calculate the angles mathematically. Two different equations were used to calculate the angles depending on the angle configuration to be calculated, i.e., whether the angle depends on two or three key points. For example, the head inclination angle is calculated using two points (head and shoulders-midpoint) that determine this inclination concerning the ground. Differently, if the angle to be calculated corresponds to elbow flexion, it is necessary to use three key points (elbow, shoulder, and wrist).

The following equations were used to calculate the flexion and inclination angles:

$$\alpha = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) * \frac{180}{\pi} \qquad \text{Eq. (1)}$$

$$\alpha = \{\arctan\left(\frac{z_2 - y_2}{z_1 - y_1}\right) - \arctan\left(\frac{x_2 - y_2}{x_1 - y_1}\right)\} * \frac{180}{\pi} \qquad \text{Eq. (2)}$$

Equation 1 expresses the inclination angle between two points, where y1 and y2 are the coordinates on the vertical axis of point 1 and 2, while x1 and x2 are the coordinates on the horizontal axis. The arc tangent results in the radians of inclination of both points, so it is necessary to convert them to degrees by multiplying by 180 and dividing by ϖ.

As in Equation 1, Equation 2 shows the coordinates of the points on both axes, with the difference that a third point is added to calculate angles between three points.

The calculated angles are described as follows in Table 1.

Table 1. Calculated angles per video-frame.

1. Head inclination.
2. Torso inclination.
3. Left arm inclination.
4. Right arm inclination
5. Left forearm inclination.
6. Right forearm inclination.
7. Left thigh inclination.
8. Right thigh inclination.
9. Left leg inclination.
10. Right leg inclination.
11. Neck flexion.
12. Left armpit flexion.
13. Right armpit flexion.
14. Left elbow flexion.
15. Right elbow flexion.
16. Left hip flexion.
17. Right hip flexion.
18. Left knee flexion.
19. Right knee flexion.

Figure 4 shows an example of a skeleton obtained from the CNN KeyPoints, where the respective key points are joined by vectors.

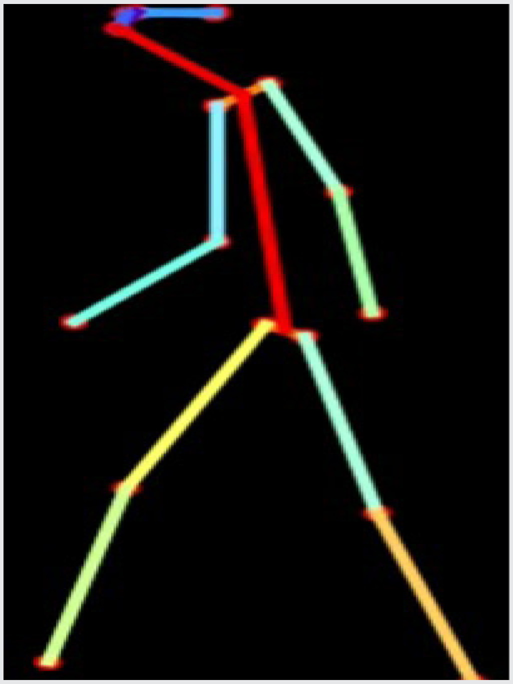

Figure 4. Skeleton with key points detected by KeyPoints CNN. Source: own elaboration.

Also, 15 data matrices are generated corresponding to each key point where the three variables of each point are stored in each video frame ordered from the first to the last. The matrix will have a size of 3 x n, where n is the extracted frames number. In the same way, 19 vectors are generated, where each angle calculated from the first frame to the last is stored.

Finally, once data vectors and matrices are created, a CSV file is generated for each matrix and vector, as well as a calculated angles graph to observe the behavior.

RESULTS

Each recorded video was between 2 and 4 seconds long, containing between 60 and 120 frames. Figure 5 shows a video frame of one recorded test subject.

Feature extraction times vary according to the equipment used and its configuration. Table 2 shows the approximate average milliseconds it takes each device to perform the extraction. In the case of units 1 and 2, the CNNs process the images in series, i.e., once one CNN finishes, the other continues. In contrast, in unit 3, the images are distributed in parallel, and both CNNs work simultaneously. Because of this, unit 3 boosts its performance and decreases the image processing time to less than half a second for each received image. Therefore, in unit 3, a video can be processed between 27.6 and 55.2 seconds.

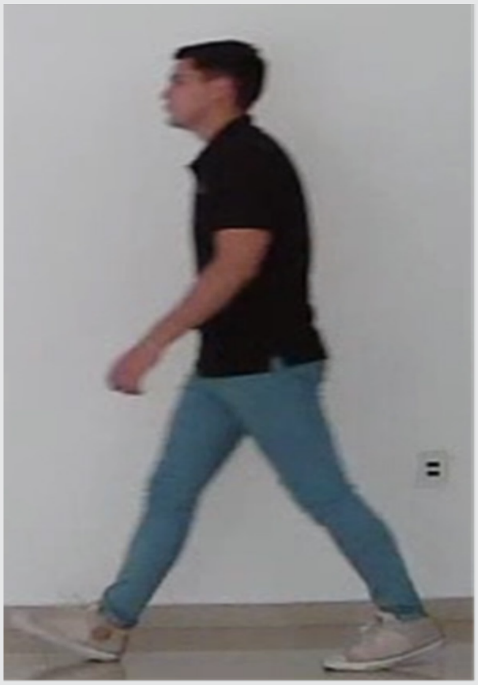

Figure 5. Video sequence frame example.
Source: own elaboration.

DensePose CNN was responsible for extracting the silhouettes from the video sequences. Once all the frames were extracted, a new AVI format video was generated for storage. Figure 6 shows a video frame with the silhouette of the test subject and without the background.

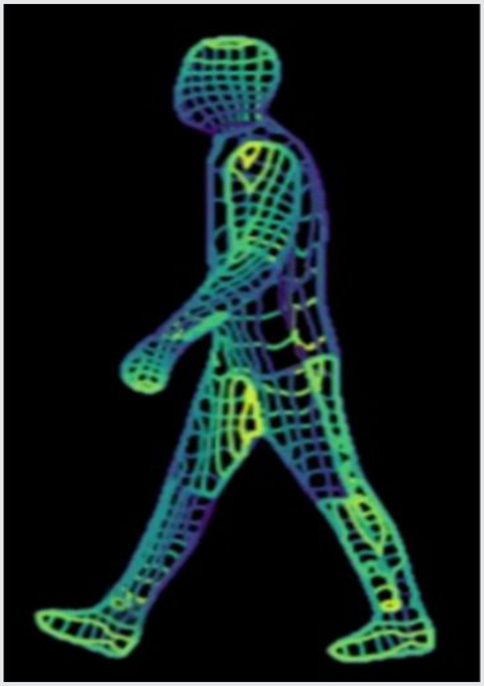

Figure 6. Silhouette extracted from a video sequence frame.
Source: own elaboration.

Table 2. Configurations used for extraction and their corresponding times per frame.

ID	Unit	Time (ms)	Execution
1	iMac Quad Core Intel i5 16 gb RAM 2,9 GHz	1004	Series
2	CPU Quad Core Intel i5 32 gb RAM 3,4 GHz	500	Series
3	CPU Quad Core Intel i5 32 gb RAM 3,4 GHz + 2x GTX 1080	46	Parallel

KeyPoints CNN extracted the key points in each video frame, which in turn served to obtain the inclination and flexion angles of the different human body parts. These angles calculated for each frame generated a graph representing the changes from frame to frame of the video. Figures 7 and 8 show the inclination and flexion graphs, respectively.

CONCLUSIONS

Security is an area of current technological boom due to the great demand for new and more innovative security systems. Time is an essential factor in preventing or identifying aggressors in certain areas. The RAH is a tool that achieves security objectives in various scenarios. A fundamental aspect of the HGR is the extracting features from the user's walk. This process can consume most of the time needed for recognition.

In the present work, a methodology was implemented capable of extracting features in the two main HGR approaches in times that do not exceed 50 milliseconds per video frame. This time allows a proper classification system to design real-time HGR systems, i.e., systems that can identify people in a short time after being captured on video. In addition, a sample of the features is provided to be used for training, visual aids, or any purpose related to HGR.

The source code and hardware configurations used to provide the possibility to replicate the experiments developed in this work to provide a new methodology related to the HGR. The repository with the video inputs, code, video outputs, key points CSV, angles CSV, and plots are available in 1.

Future work

After feature extraction, it is necessary to perform data dimensionality reduction to improve the classification process of these features.

Classification is the last step in determining a person's identity through gait. The most important features of a person's gait are used to train neural networks to classify them and output the person's identity.

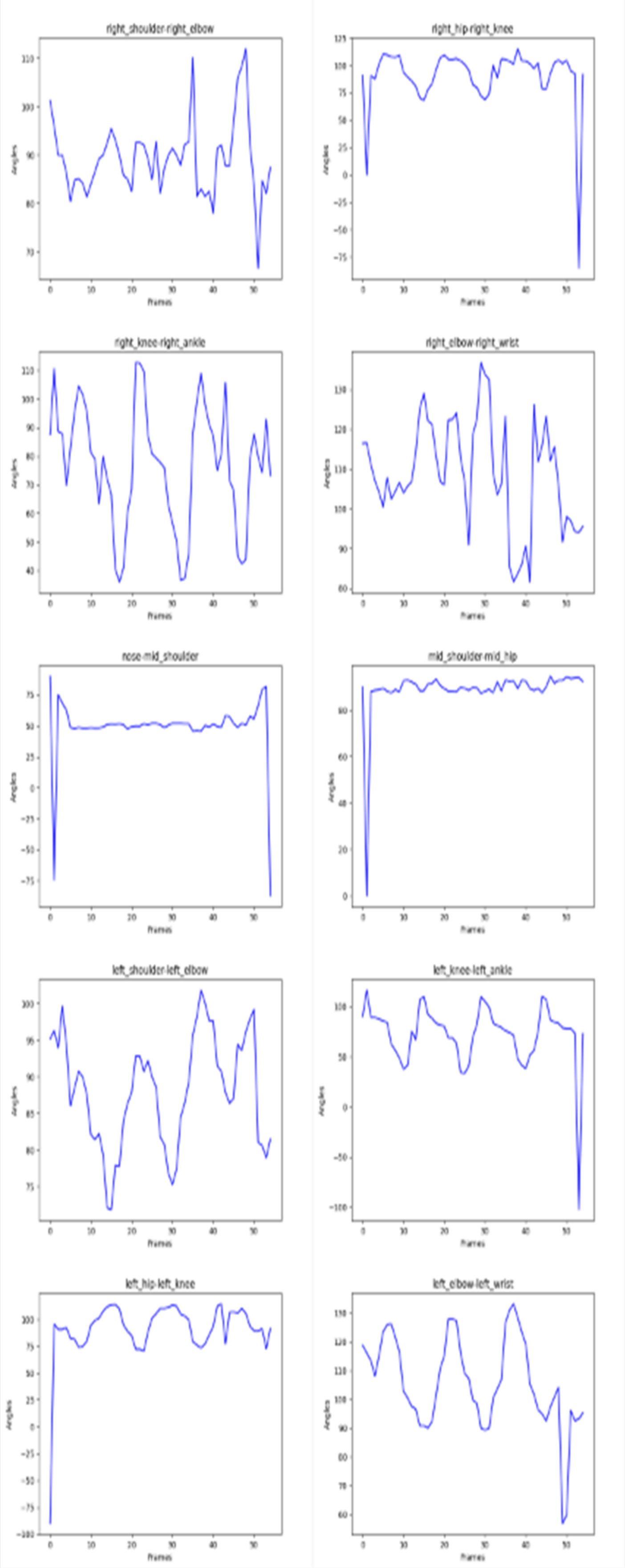

Figure 7. Inclination angles from a video sequence. Source: own elaboration.

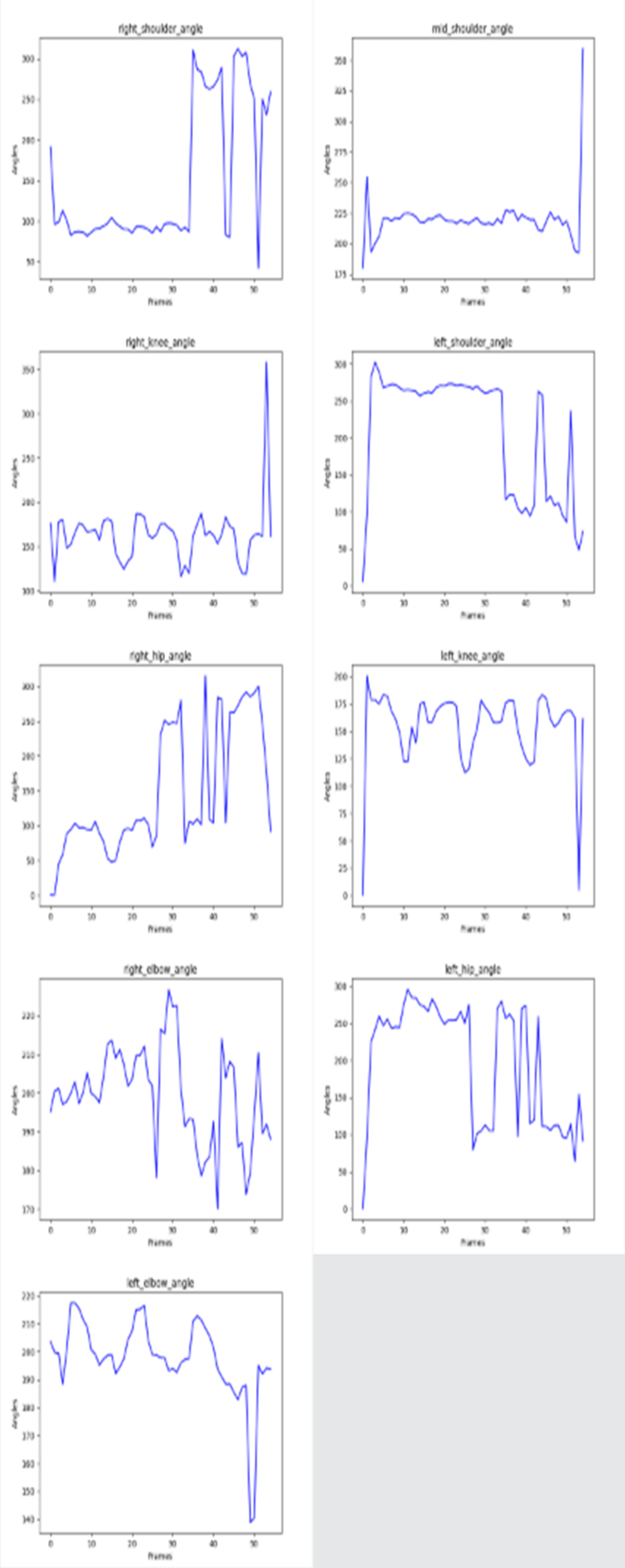

Figure 8. Flexion angles from a video sequence. Source: own elaboration.

BIBLIOGRAPHY

[1] UNODC. (2018), Burglary. Obtenida el 03 de enero del 2023, de la página electrónica: https://dataunodc.un.org/data/crime/burglary.

[2] Yang, W., Hu, J. & Wang, S. (2014). A Delaunay quadrangle-based fingerprint authentication system with template protection using topology code for local registration and security enhancement. IEEE transactions on Information Forensics and Security. 1179-1192.

[3] Bud, A. (2018). Facing the future: The impact of Apple FaceID. Biometric technology today. 5—7.

[4] Yan, Z. & Zhao, S. (2016). A usable authentication system based on personal voice challenge. International Conference on Advanced Cloud and Big Data (CBD). 194—199.

[5] Thavalengal, S., Bigioi, P. & Corcoran, P. (2015). Iris authentication in handheld devices-considerations for constraint-free acquisition. IEEE Transactions on Consumer Electronics. 245—253.

[6] Sidiropoulos, G., Kiratsa, P., Chatzipetrou, P. & Papakostas, G. (2021). Feature Extraction for Finger-Vein-Based Identity Recognition. Journal of Imaging. 89.

[7] Al-Waisy, A., Qahwaji, R., Ipson, S., Al-Fahdawi, S., Nagem, T. (2018). A multi-biometric iris recognition system based on a deep learning approach. Pattern Analysis And Applications. 21, 783-802.

[8] Moreno, M. (2008). Reconocimiento del andar humano basado en ensamble de clasificadores utilizando silueta y contorno. Instituto Nacional de Astrofísica, Óptica y Electrónica.

[9] Lin, B., Zhang, S., & Bao, F. (2020, October). Gait recognition with multiple-temporal-scale 3d convolutional neural network. In Proceedings of the 28th ACM international conference on multimedia. 3054-3062.

[10] Bouchrika, I. (2018). A survey of using biometrics for smart visual surveillance: Gait recognition. Surveillance In Action. 3-23.

[11] Han, J. & Bhanu, B. (2005). Individual recognition using gait energy image. IEEE Transactions on Pattern Analysis and Machine Intelligence. 316-322.

[12] Güler, R. A., Neverova, N., & Kokkinos, I. (2018). Densepose: Dense human pose estimation in the wild. In Proceedings of the IEEE conference on computer vision and pattern recognition. 7297-7306.

[13] Yuxin W., Alexander K., Francisco M, Wan-Yen L. & Ross G.. (2019). Detectron2. https://github.com/facebookresearch/detectron2.

[14] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition. 770-778.