International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-021-01372-y

ORIGINAL ARTICLE

Hand gesture recognition using machine learning and infrared information: a systematic literature review

Rubén E. Nogales[1,2] · **Marco E. Benalcázar**[1]

Received: 22 August 2020 / Accepted: 1 July 2021

Abstract
Currently, gesture recognition is like a problem of feature extraction and pattern recognition, in which a movement is labeling as belonging to a given class. A gesture recognition system's response could solve different problems in various fields, such as medicine, robotics, sign language, human–computer interfaces, virtual reality, augmented reality, and security. In this context, this work proposes a systematic literature review of hand gesture recognition based on infrared information and machine learning algorithms. This systematic literature review is an extended version of the work presented at the 2019 ICSE conference. To develop this systematic literature review, we used the Kitchenham methodology. This systematic literature review retrieves information about the models' architectures, the implemented techniques in each module, the type of learning used (supervised, unsupervised, semi-supervised, and reinforcement learning), and recognition accuracy classification, and the processing time. Also, it will identify literature gaps for future research.

Keywords Gesture recognition · Infrared information · Machine learning · Systematic literature review

1 Introduction

Gestures are considered as a natural expression of the human body and are used to communicate with other people [1, 2]. The gesture most popular for communicating is the hand gesture. It is recognized as hand movement in a given time. It is estimated that two-thirds of the communications execute using signs [3]. There are two types of hand gestures:

Static—Consists of a well-defined position of the hand in a given moment.
Dynamic—It is a sequence of movements in a lapse of time.

The outputs of hand gesture recognition systems are using in: sign language communication [4, 5], human–machine interaction [6, 7], human–robot interaction, and virtual reality [8]. In this context, hand gesture recognition is a problem composed of two subproblems: feature extraction and pattern recognition. Hand gesture recognition consists of mapping an input of a set with a set of labels, where a label denotes a gesture to be recognized. Also, it is necessary to identify the instant of time when the movement is executed [9].

Implementing a hand gesture recognition system consists of joining different modules such as data acquisition, preprocessing, feature extraction, classifiers, and post-processing. The classification module could design using machine learning, especially when the problem is very difficult or even impossible to find a mathematical model (i.e., probability distribution). Finding a mathematical model requires knowing the dynamics of the problem and its behavior. Therefore, finding a mathematical model that describes hand gesture recognition with high precision is a difficult task.

The sensors used to acquire data in hand gesture recognition systems include gloves, RGB cameras, Myo Armband, brain-computer interface (BCI), and infrared sensors. The use of gloves can be uncomfortable for executing hand movements because it is a foreign body. The use of RGB cameras involves dealing with segmentation problems, finger occlusions, and lighting changes [10–13]. The Myo Armband and BCI sensors face noise produced by sensors

✉ Rubén E. Nogales
ruben.nogales@epn.edu.ec; re.nogales@uta.edu.ec

Marco E. Benalcázar
marco.benalcazar@epn.edu.ec

[1] Escuela Politecnica Nacional, Quito, Ecuador

[2] Universidad Tecnica de Ambato, Ambato, Ecuador

Published online: 12 July 2021

or environment, the variation of signals in electrodes due to sweating, and electrode donning-doffing (put on and take off the sensors) [14]. In this context, infrared sensors are an alternative solution for implementing hand gesture recognition because these devices do not present the problems described above.

To develop this systematic literature review uses the Kitchenham methodology. It presents three phases, *planning*, *conducting*, and *reporting the review* [15].

In the *planning phase*, we identified the need to conduct a literature review assessing whether there are any systematic reviews of hand gesture recognition using machine learning and infrared information. Besides, it develops a protocol for the literature review; this protocol helps us avoid bias in the research. The protocol identifies the research questions and defines the strategies for selecting the primary studies, such as scientific database, selection criteria, and quality assessment.

The *conducting phase* consists of identifying the primary studies about the research problem using the search strings. Next, we select studies based on selection criteria, from these studies carried out data extraction and monitoring progress, finally, presents a data synthesis.

In the *reporting phase*, the work presents an interpretation of data and discusses the generalizability of the review's conclusions and limitations.

1.1 Contribution

In this work, we will carry out a scientific literature review that allows us to define the hand gesture recognition problem's progress using infrared information and machine learning algorithms. It will also allow us to determine the architectures used by other researchers, the techniques they use in the respective modules, the devices they use, the algorithms' execution times, and the reported accuracy. As a result, this study will present an analysis of previous studies and identify trends and gaps for new studies.

2 Planning phase

This phase defines if exist previous reviews about the problem. Besides, it presents a protocol by means to face the phenomenon of study.

2.1 Review of reviews

A review of systematic literature reviews is carried out to know if other researchers have already addressed the problem and if that is the case, retrieve the following items: author's names, paper title, year of publication, objectives, sources of the primary studies, inclusion criteria, exclusion criteria, criteria for quality assessment, data extracted, and findings, to determine its progress [15].

For this study, we used search strings that connect the following words: systematic literature review, state of the art, review, survey, hand gesture recognition, tracking, machine learning, and infrared. The works reviewed in this section include papers published in journals and conferences from the following databases: IEEEXplore, ACM Digital Library, Willey Online library, Science Direct, and Springer. For avoiding bias also searched in google scholar. Then the following search string is used:

(1) *(((hand AND gesture AND (recognition OR tracking)) AND "machine learning") AND (infrared OR ("infrared information"))) AND ((("systematic literature review") OR ("state of the art") OR review OR survey).*

We obtain 13,374 papers using the search string and databases described above. As a result, there is only one coincidence, but this paper is the previous study of this work. Besides, the search presents 20 articles similar to the study's phenomenon, as is described in Table 1. In this context, none of these papers addresses the aspect of hand gesture recognition using machine learning and infrared information. However, seven reviews conduct hand gesture recognition using machine learning and vision algorithms, which are part of this section.

2.2 Related works

The study of related works consists of describing common topics to the research. In this context, it is necessary to identify criteria such as (a) describe objectives, (b) sources of primary studies, (c) inclusion criteria, (d) exclusion criteria, (e) quality evaluation criteria, (f) data extraction, (g) findings, (h) methodology, (i) research questions.

Reference [10] presented three research questions that address the methods, applications, and environmental

Table 1 Result of the search string of review of reviews

Scientific databases	Search string		
	Golden search string	Related works	Total results
IEEE Xplorer	0	1	1
ACM digital library	0	1	366
Willey	0	2	118
Science Direct	0	5	244
Springer	1	3	545
Google scholar	0	8	12,100
Total	1	20	13,374

conditions used for hand position and testing gesture recognition. For this purpose, it uses five depth sensors. However, only the Kinect sensor was studied in-depth, while briefly mentions the XtionPro and stereoscopic cameras. This study presents two hand-localization methods, such as hand segmentation and hand tracking. For hand gesture classification presents two methods, Hidden Markov Models (HMM) and Artificial Neural Networks (ANN). However, the authors present these as recognition methods. This survey does not mention what methodology they use, how they obtain the primary studies, inclusion and exclusion criteria, which assessment criteria used, how were data extracted, and how data synthesis was. Finally, they do not critical of the primary studies (Table 2).

In Ref. [3], the authors described the static and dynamic hand gesture recognition. For this purpose, they used as a classifier HMM, ANN, Eigenspace, Curve fitting, and Dynamic Programming/Dynamic Time Warping. Besides, this study described the use of supervised and unsupervised learning and analyzed RGB-D cameras. However, this study does not mention the methodology, inclusion and exclusion criteria, and the quality of criteria assesses.

In Ref. [16], the authors presented a study about seven patients-monitoring applications on vision-based such as fall detection, action, activity monitoring, epilepsy monitoring, vital signs monitoring, and facial expression monitoring. Each of the applications presented comprehensive studies about technologies used, such as multi-camera, monocular digital, infrared time-of-flight camera systems, and bio-inspired vision sensor-based systems. However, it did not present the databases that retrieved information from the primary studies. In the same way, it did not present the inclusion and exclusion criteria and quality assessment criteria.

In Ref. [17], the authors presented a systematic literature review for understanding 3D mid-air hand gesture recognition. Also, it presents three research questions to define, classify, and identify mid-air gestures. However, the authors did not present gaps, future research, or the analysis of papers included in this study.

In Ref. [18] presented a theoretical of gestures description types such as iconic, metaphoric, beats, and cohesive gestures. Moreover, mentioned that the gestures have four phases: preparation, pre-stroke hold, stroke, and retraction, supposed meaning the gesture is in the stroke phase. Also, it presented the relationship between gestures and semantics and described the segmentation phase. Similarly, presented data representation strategies, computational techniques or strategies used in data analysis, and metrics employed to evaluate the results. Nevertheless, this paper did not mention in which databases the search was carried out the primary studies, did not mention the inclusion and exclusion criteria and quality assessment criteria used for evaluating the papers.

In Ref. [19], the authors presented four research questions focused on technologies, techniques, algorithms, application domains, and hand gesture recognition commercial software based on vision. For answering these research questions, the authors follow the Kitchenham methodology, and this study analyzed two approaches: sensor-based and vision-based. This study presents a comparison between the main vision-based techniques. Also, it presents a description and comparison between different algorithms used hand gesture recognition vision-based. The authors briefly describe application fields and present gaps such as real-time, cost factor referent to software and hardware, and background challenges between others. However, the comparison between these two approaches is superficial and is not critical. In the same sense, the authors have not been critical of the primary studies and the primary reviews.

In Ref. [20] presented a comprehensive survey, new techniques, topics, methods, sub-methods, and soft computing problems for image segmentation. Furthermore, Fuzzy logic, ANN, and Genetic algorithms are the core of soft computing. They did not mention which databases they use for searching primary studies. It did not show how the primary studies are selected. However, each paper evaluated parameters such as testing protocol, testing regime, performance indicators (accuracy, robustness, sensitivity, adaptability reliability, efficiency), performance metrics of the algorithms, and image databases used for testing algorithms. The paper described the most widely

Table 2 Result of the search string of review of reviews

References	Search string								
	Year	a	b	c	d	e	f	g	h	i
[10]	2012	X	–	–	–	–	X	X	–	X
[3]	2015	X	X	–	–	–		X	–	–
[16]	2015	X	–	–	–	–	X	X	–	–
[17]	2016	X	X	X	X	X	–	–	X	X
[18]	2016	X	–	–	–	–	X	X	–	–
[19]	2018	X	X	X	X	X	X	X	X	X
[20]	2018	X	X	X	X	–	X	X	–	X

used databases for image segmentation. The paper lets future available works. However, it did not mention the methodology used.

As we can see, the scientific literature does not show that exist a systematic literature review concerning hand gesture recognition based on infrared information and machine learning techniques. However, it presents models for the recognition and classification of static and dynamic hand gestures. These models are based on infrared information, depth and color images [6, 21–26]. Also, it presents models that work in real-time, which is a challenge due to hardware restriction and processing time [21, 23, 27, 28].

2.3 Review protocol

The systematic literature review's main goal is to retrieve information of primary studies that other researchers generated, identifying trends, gaps for new studies, and present the analysis of previous studies about hand gesture recognition using machine learning and infrared information. In this context, the review protocol permits the researchers to avoid bias in selecting primary studies and present the wrong results. In this section, the methodology proposes developing topics such as *research questions, search strategies for primary studies, selection of relevant studies, quality assessment of the selected studies, and discussion of the results* [15].

2.3.1 Research questions

The definition of the research questions is one of the essential activities in the systematic literature review. These research questions will address the extraction of information from primary studies. In this context, to define the research questions, the methodology proposes to establish the *population*, *intervention*, and *outcomes*.

2.3.1.1 Population The population is an excellent collection of samples; these are the focus of a scientific query, so the research aims to resolve the population issues. In this research, the population will be *all instances of the gestures generated by the hand*, such as the kind of gestures, number of recognized gestures, and number of samples retrieved.

2.3.1.2 Intervention Intervention is the technology that allows us to provide a solution for a specific problem. There are different technologies related to hand gesture recognition, and for this study, the interventions are models based *on machine learning and the infrared*. The architecture of the models based on machine learning can be the combination of the following modules*: Data acquisition, Preprocessing, Feature extraction, Classification, and Postprocessing*.

Combining these modules will allow us to obtain different results so that the performance will vary according to it. Similarly, the infrared sensors will permit to get the type of information that represents the problem.

2.3.1.3 Outcomes These are the possible outputs obtained after applying different architectures, techniques, and data specified in the intervention. In this context, the models' results based on machine learning could be improving, then in this study, the outcomes will be: Recognition accuracy and speed of processing.

With the exposed above, the present systematic literature review proposes the following research questions.

General research question

- Which is the state of the art of the existing models for hand gesture recognition that uses machine learning and infrared information?

Specific research questions

- What is the architecture of the proposed models for hand gesture recognition based on Machine Learning and Infrared information?
- What are the protocols, types of sensors, and types of the dataset used to develop hand gesture recognition models based on machine learning and infrared information?
- What types of learning (supervised learning, semi-supervised learning, unsupervised learning, or reinforcement learning) have been used to train hand gesture recognition models with infrared information?
- What are the processing time and recognition accuracy of hand gesture recognition models that use machine learning and infrared information?

2.3.2 Strategy for searching primary studies

The present systematic literature review is a process that retrieves valid information from primary studies of hand gesture recognition using machine learning techniques and infrared information.

In this section, we present the scientific databases for extracting the primary studies and the keywords for building search strings. For this study, the review of primary studies will be in online databases, as shown in Table 3.

The topics used for building the search string are *hand gesture, hand gesture recognition, hand tracking, hand poses recognition, machine learning, machine learning techniques, machine learning algorithms, infrared, and infrared information*.

The use of these is because it covers a broad spectrum of the proposed research questions, and for minimizing

Table 3 Scientific database online

Databases	URLs
ACM	http://dl.acm.org/
IEEE Xplorer	http://ieeexplore.ieee.org/
Science Direct	http://sciencedirect.com/
Springer	http://link.springer.com/
Willey	http://onlinelibrary.wiley.com/

the risk of unreviewed works, attaches synonyms. These synonyms are defined by reading related articles.

Gestures are natural movements of the human body. It transfers certain valid information to maintain an interaction with the environment. In computer science, these gestures generally come from the movements of the face or hands, and our field of study is the information generated by hand gestures.

In computer science, modeling the gestures uses mathematical algorithms, but some of these events generated by hand gestures can be very difficult or impossible to model. In this sense, it is necessary to model these events using machine learning algorithms, knowing that machine learning algorithms use techniques that learn from data.

The acquired data are through external devices such as cameras, gloves, among others. Some of these devices' present complications, as explained in the introduction. However, to minimize these problems, it is necessary to use infrared information.

In this context, the search strings are build combining these topics, its synonyms, and connectors as conjunction AND, and disjunction OR, as following shows.

((((hand AND (gesture OR poses)) AND recognition) AND "machine learning") AND infrared)
((((hand AND (gesture OR poses)) AND recognition) AND ("machine learning" OR "machine learning techniques" OR "machine learning algorithms")) AND (infrared OR "infrared information"))
(("hand tracking" AND ("machine learning" OR "machine learning techniques" OR "machine learning algorithms")) AND (infrared OR "infrared information"))

Additionally, to extract the greatest amount of valid information exposed in the scientific literature, a general search string for the problem in question is presented.

((((((hand gesture) OR (hand poses)) AND recognition) OR (hand tracking)) AND machine learning) AND ((infrared) OR (infrared information) OR (Leap Motion) OR (Kinect))).

2.3.3 Procedure for relevant studies selection

The search for primary studies will be carried out only in the databases shown in Table 3. From this search, only chose articles presented in journals and congresses. The process will develop as follows:

1. To enter each search string inside each database, shown in Table 3.
2. Constrain the search of articles between 2015 and 2019 and articles of congress and journals.
3. Select all studies that in the title presents:
   a. Hand gesture recognition
   b. Hand poses
   c. Hand tracking
   d. Any three topics with any machine learning algorithms
   e. Any above topics with any infrared device
4. If the title does not mention any of the topics in point three, we search in the abstract, keywords, and the conclusions, whether the article mention infrared sensors, machine learning algorithms, or hand gesture recognition.
5. The chosen articles register in a spreadsheet.
6. The second time evaluates the selected articles applying inclusion and exclusion criteria shown in Table 4.
7. Register the chosen articles from numeral six in a second spreadsheet.
8. To evaluate the chosen articles for the third time is used a Likert scale, which determines a scientific validation, as shown in Table 5.
9. Get the information from the selected articles in numeral eight to answer the research questions. Below are the items that allow us to complete this task.

Model structure

- Data acquisition, pre-processing, feature extraction, classifier, and post-processing.

Dataset

- Classes number, types of gestures, type of input, type of hardware, data origin, sample number, sampling frequency, subjects number, types of subjects, and acquisition protocol.

Training

Table 4 Inclusion and exclusion criteria

Inclusion	Articles that use machine learning and infrared information for hand gesture recognition
	Only articles from databases shown in Table 3
	Only articles of congresses and journals
	Peer-reviewed items
	Works that present models or that compare model
	Publications between January 2015 and December 2020
Exclusion	Articles that are not related to hand gesture recognition
	Articles that do not use infrared information and machine learning for hand gesture recognition
	Articles with years of publication earlier than 2015
	If the publications do not define population, intervention, and outcomes (accuracy and time)
	All articles that do not be in the English language
	Works that present only applications and that do not propose a model

Table 5 Assessment quality criteria of the articles using a likert scale weighting

Quality criteria	a	b	c	d	e
The findings are credible				0.5	
The findings are important				0.5	
The research brings new knowledge			0.25		
The evaluation address well its original aims and proposal				0.5	
The scope of research let new researches				0.5	
The basis for evaluating the result is clear				0.5	
The research design is defensible			0.25		
The sample design, target selection of classes, is well document		− 0.5			
The data collection was well carried out		− 0.5			
The approach, formulation and, analysis of the problem has been adequately carried out				0.5	
The diversity of perspective and context has been explored (related work)				0.5	
The links between data, interpretation, and conclusions clear			0.25		
The reporting is clear and coherent			0.25		
The theoretical contributions, the perspectives, the values that the research leaves are clear				0.5	
The research process has been adequately documented				0.5	

- Samples number for training, validation, and testing, machine learning algorithms.

Time/speed

- Real-time measured in time unit, growth operations, or growth of memory.

Accuracy

- Detailed results, variability, compare results, analyze results, and evaluation protocol.

3 Conducting the review phase

This phase consists of finding the primary studies related to the research questions and extracting the most relevant information from these studies. For avoiding the impartiality of the selection, it is necessary to follow strictly and systematically the steps described in the protocol. Besides, we select the primary studies based on the inclusion and exclusion criteria and the quality validation criteria.

3.1 Inclusion and exclusion criteria

This section defines the inclusion and exclusion criteria. Besides, it determines whether it includes or excluded the primary studies in the systematic literature review. In this sense, selecting the most relevant studies depend on the clarity of its definition and its application.

The topics of hand gesture recognition above exposed, the articles reviewed for pairs, and the articles published in journals and congress define the inclusion criteria. Basically, three rules define the exclusion criteria. These are: *Poorly*, *potentially justified reasons for excluding,* and *strongly*.

The poorly justified reason is the articles that are out a range of dates and articles written in other languages different from English.

The *potentially justified reason for excluding* is when not define models only present applications i.e. [29, 30].

Finally, *strongly* justified reasons for excluding are all articles that do not define population, intervention, and outcomes. It details the inclusion and exclusion criteria in Table 4.

3.2 Quality assessment

Finally, after selecting the articles using inclusion and exclusion criteria, they will be evaluated using a Likert scale. The Likert scale consists of defining quality assessment criteria and evaluation criteria for quality weighting. The criteria for quality assessment evaluates the methodology, the obtained results, how they are presenting reports, among others as shown in Table 5, while the weighting criteria are: (a) strongly disagree, (b) disagree, (c) neither agree nor disagree, (d) agree, and (e) strongly agree.

The researcher evaluates each quality assessment criteria reported in the papers, giving a weighting of -1, -0.5, 0.25, 0.5, and 1 to the quality assessment criteria and quality weighting, respectively. We choose the values of weights because if they exist, the same amount of evaluation criteria in a and b as d and e, the summation zero $\sum a + b + d + e = 0$, this leaves the analyzed article in the middle of the quality with the summation of $\sum c$.

In the case that the summation of the weights between *a, b, d, e* are not zero $\sum a + b + d + e \neq 0$, the summation of c is taken as the threshold $\tau = \sum c$, and for an article to be part of the review, the value of the total summation of the weights must be equal to or greater than the threshold $q_s = \sum a + b + d + e \geq \tau$.

In this context, the studies that form part of the systematic literature review comply with a threshold equal to or greater than the sum of the values of the evaluation criteria of literal c. Table 5 shows an example of evaluation where (a) strongly disagree, (b) disagree, (c) neither agree nor disagree, (d) agree, and (e) strongly agree.

In this work, following steps 1 to 4 described in the previous section's protocol, 1174 items were found in indexed databases. Also, to avoid biases of including only positive results, the ArXiv database was included and found 231 articles. After this step, it applies the inclusion and exclusion criteria in two phases:

In the *first phase,* it constrains the search from 2015 to 2020. The articles that in their titles contain the keywords defined above (point 3) were selected. If the keywords do not find, the search will be in abstracts and conclusions. In this phase, we obtained 203 articles: 56 from IEEE, 31 from ACM, 44 from ScienceDirect, 47 from Springer, six from Willey, and 19 from ArXiv.

In the *second phase*, we excluded all articles that were not in English and articles that present only applications. Besides, it includes articles that present population, intervention, and outcomes.

In this phase, we obtained 69 articles: 27 from IEEE, 7 from ACM, 18 from ScienceDirect, 14 from Springer, two from Willey, and one from ArXiv.

Besides, the quality assessment criteria were applied. These secure that the selected articles have the best quality for adding to the study. As a result of applying these criteria, we obtained 44 articles: 12 from IEEE, five from ACM, 15 from ScienceDirect, 11 from Springer, and one from Willey. From the works selected are 21 articles in Journals and 19 in congress.

Finally, the information is retrieved; the criteria used to retrieve information is shown in protocol point 9, the information extracted is arranged in a spreadsheet. To secure that the information retrieved is according to the quality assessment criteria, it discussed the information with the second author of this paper.

3.3 Results

For obtaining information about the primary studies, we defined a generic model. This model will permit the evaluation of the architecture proposed for different works in the scientific literature. The generic model is composed of:

Data acquisition It is a set of methods that permit to retrieve data of devices used in the proposed works.
Pre-processing The use of methods that give an input signal permits obtaining a different signal; the purpose is to increase the model's chance of accuracy.
Feature extraction It is a set of descriptors that represent very well a signal returned by the pre-processing module.
Classifier It is a set of algorithms that, given an input signal in the form of a vector, can return a label of a set of labels previously established.
Post-processing This module consists of improving the response given by the classifier.

This section's main goal is to answer the research questions described above and extracted the most relevant information.

In Ref. [31], a dynamic gesture recognition system based on short term and long term memory networks (LSTM) and convolutional neural networks (CNN) is presented. The system evaluates six classes; For evaluation, the system collects data with the LMC. The data are 3D spatial positions and finger velocity. To build the dataset, the authors keep single-take records of the six gestures. To label these data, they perform a manual signal segmentation, aided by a video recording. Also, they propose to train an algorithm based on LSTM to label new data automatically. To improve the labeling of the data, they generate post-processing, in which it is determined if there are frames smaller than ten between the gesture, they are put together within the signal of the same gesture. Otherwise, it is labeled as a transient frame between gestures. In this sense, the proposed system comprises two modules, the recognition module, and the classification module. These data feed the classifier. If the classification is with LSTM, the authors mention that they use the same gesture detection architecture. Also, they train the model using a CNN, the output of the network feeds a softmax. To train the model, they use the cross-validation technique and report an accuracy of 98.4% and 125 ms. However, it does not present a protocol for data acquisition, accuracy, and time measurement. It does not mention how many people created a data set.

In Ref. [32] proposes a 16 gesture recognition system in real-time using the Kinect sensor. The paper uses depth images because this type of image solves the problem of background, illumination changes, or overlapping images. For the preprocessing, they present the calculation between the maximum and minimum pixels of the depth image, which allows the mobility of the subject performing the gesture. They also use the floodfill algorithm to find the connected regions that allow defining the hand. They present SIFT and SURF as feature extraction techniques. SIFT is a progressive convolution process between images with different sizes and a Gaussian kernel, followed by subtraction of each successive image. Once it locates the key points based on their stability, the algorithm assigns an orientation to each detected key point by collecting gradient directions and magnitudes around it. Finally, the algorithm generates a highly characteristic fingerprint for each Keypoint. SURF is very similar to SIFT, and the difference is that SIFT has a 64-feature descriptor and SURF has 128 features. They train the model using a support vector machine (SVM) based on linear and radial kernel. The dataset is composed of 8000 images. To train the model, they use 4800 images, 1600 to validate, and 1600 to test. They report an accuracy of 98% using the SURF extractor, while when using SIFT, they report 91%, with an average of 0.30 s with SIFT and 0.12 with SURF. They also briefly detail a data acquisition protocol. They do not mention the number of subjects to construct the data set.

In Ref. [33] presents the development of a rehabilitation platform, however, this study is taken for the systematic literature review, because it also presents a classification and recognition model of eight static and dynamic gestures. The dataset is building using 30 subjects, and each one repeats 100 times the gesture. In this paper, LMC acquired the data with a sampling frequency of 150 frames/s, and these data are the position and the speed of the fingers. In the data acquisition protocol, it mentioned that the subjects of the experiment had not suffered any injury. As pre-processing, they use a smoothing function. They use division by windows as a feature extraction technique with $w = 20$. Also, compute the mean, the distance, and angles between fingertip and palm center. As classifier of the static gestures, it uses discriminant analysis (DA) and SVM. Simultaneously, the dynamic gestures use hidden Markov models (HMM), reporting accuracy of 99.09% and 98.76%, respectively. The evaluation of the rehabilitation system reports an accuracy of 80%. It does not report processing time.

In Ref. [34], developing a model that recognizes 28 letters of the Arabic alphabet is presented. In this sense, it recognizes static and dynamic gestures. The devices used for data acquisition are the LMC and the Kinect, with the Kinect it acquires depth images, while the LMC acquires velocity and orientation data. To obtain a consistent and unified dataset, the authors adjust the data from the two sensors to a standard length in millimeters. To build the dataset, a protocol is presented and was constructed with 20 subjects between 15 and 40 years old. As a preprocessing technique, they use principal component analysis (PCA) to adjust the acquired data's length because the hand's representation in one shot maybe 120 points. In contrast, in another shot, it may be 60 points. Besides, they eliminate the redundancy of data and data considered not necessary. They construct the feature vector from the normalized finger lengths. They also use the angles between point (i) of the fingers and the x, y, and z axes, between the wrist and the hand orientation data, between the joints presented in the depth images. These data feed the SVM classifier with a Gaussian kernel. To train the model, it uses 1121 samples and validates 280 samples. It reports an accuracy of 93% in training and 86% in testing. It does not report post-processing nor processing time.

In Ref. [35] proposes the recognition of sign language and semaphoric hand gestures. They use 30 gestures, 18 static, and 12 dynamics. They retrieve spatial coordinate data using CML, particularly the joints' angles involving the distal, intermediate, and proximal phalanges for the little finger, ring, middle, and index fingers, while for the thumb, the metacarpal. To train and test the model, they use the public SHREC database. However, they also create a proprietary

dataset. This dataset consists of 1200 samples, acquired from 20 subjects between 20 and 28 years old, 15 males and five females. They presented a data acquisition protocol and acquired the sample for 5 s, at a sampling rate of 200 Hz. The dataset does not present data preprocessing. However, they present a feature vector composed of the angles formed by the distal and intermedial phalanges' internal angles when it comes to the thumb, the angle between the distal and the proximal phalanx. The angles between the intermediate and proximal phalanges and the angle between the proximal phalanx and the metacarpal for the thumb. Also, add spatial positions of fingertips. These features feed the classifier called deep-long-short term memory (DLSTM). The authors call it this way because it is an architecture formed by a recurrent neural network (RNN) and LSTM network. The model is trained with 780 samples and tested with 420 samples. They report an accuracy of 96.4102%, precision of 96.6434%, Recall of 96.4102%. However, they do not present a protocol for how they report the data.

In Ref. [36] presents a gesture recognition system. It recognizes the Arabic numerals (0–9) and the capital letters A and Z. In this sense, the work presents 12 dynamic gestures. The paper aims to demonstrate how easy and accurate it is to capture the data through the Leap Motion Controller and provide an effective gesture recognition method based on finger positions and hand orientation characteristics. The authors acquire the spatial positions and direction of the fingers. They build a dataset using 12 subjects and repeat each gesture 10 times, obtaining 1200 samples. The paper does not present a preprocessing module. However, it clearly presents the deterministic learning theory and mentions that they use as radial basis function the Gaussian function. The system presents two phases, the training phase, and the recognition phase. In the training phase, they present the 3D finger data acquisition, calculate the fingertip motion angles, then perform a feature selection and model the dynamic motion of the fingers. For the recognition phase, they acquire new data, calculate the fingertip motion angles, perform a feature selection and with the training result build a bank of dynamic estimators. The angles are defined by the index finger's movement in the x–y and x–z plane at the time interval t. In feature selection, they take the fingers' x and y spatial positions and the previously calculated angles. These features feed a neural network with a radial basis function. For training, they use the cross-validation technique. They report an accuracy of 95.83% with twofold and 97.25% with 10 folds. It does not report values for classification and recognition.

In Ref. [37] presents a medical image manipulation system using 11 dynamic gestures. The gestures were chosen during technical visits to hospitals in discussions with surgeons who see the need for image handling in a sterile environment. The LMC acquires the data, and the data they use to build the system is the spatial positions and direction of the fingers and palm. The dataset contains 550 samples, taken from 10 people, three men and seven women, and each gesture repeated five times. The model is composed of the data acquisition, feature extraction, and classification module. The paper does not directly mention using a preprocessing module, but they normalize the data over a range of $[-1, 1]$. To form the feature vector, they use the window splitting technique, with $w = 20$; for this, the data used are the palm center's spatial positions and fingertips. These data vectors extract the arithmetic mean, standard deviation, covariance, and root mean square. In this sense, they obtain six vectors. These vectors are concatenated and form the feature vector. These features feed the SVM classifier based on nonlinear Gaussian radial basis functions as the kernel. To train the model, they use the cross-validation technique and report an accuracy of 81%.

In Ref. [38] they present the solution to the finger occlusion problem using two LMC sensors. For the experimental demonstration of the proposal, they use three gestures and characterize them by estimating the fingertips' position, the palm, the normal and direction vectors, and the hand's rotation. In this sense, the authors train an offline classifier using an artificial hand. This is done to define the most likely position and orientation to achieve the highest estimation accuracy. These data are captured by the 2 LMCs at a sampling rate of 120 frames/s, creating a data set of 108 samples. These data are processed to analyze the hand position directly instead of examining the image in depth. The authors mention that it is necessary to calibrate the sensors at an angle of 150 degrees because the cameras' field of view can be crossed. As preprocessing, they present the transformation from a global coordinate system to a local coordinate system of the hand. A feature vector composed of x, y, and z is obtained from the palm's relative position, the normal plane, the direction values, the rotation at the Roll angle, the dot product between the normal vector and the direction, and a sensor confidence estimate value. These data feed the SVM classifier. They do not report how they train the model but report a recognition accuracy of 90.80%. However, they do not differentiate whether it is classification or recognition because they do not report any protocol.

In Ref. [39] presents a novel way of recognizing hand gestures using in-depth images and Dynamic Time Warping. The images are captured by the Kinect, with a frequency of 40 images per s. The system recognizes 55 gestures, 10 static gestures, and the American sign language. In the preprocessing module, the images are segmented and use the k-curvature algorithm to identify the palm center's fingertips, points that define the contour of the hand. It also proposes as a feature vector the relative position of the fingers about the center palm position. They considered the data as time series functions and compared with stored data using

DTW, and take as the classification value the returning time series similarity value. It does not present a training protocol but returns an accuracy of 93.9% classification. They do not mention a recognition protocol. Although it mentions real-time, the reported time is 2.9 s, and according to the literature, for it to be in real-time, the recognition time must be less than 300 ms.

In Ref. [40] proposes a method of hand gesture recognition that combines the movement trajectory with the hand-shape change. This system recognizes eight dynamic gestures based on the Kinect's depth images; the gestures are from the Chinese alphabet. They present a dataset of 800 samples, with data of 20 people, and each person repeats the gesture 5 times. The pre-processing module uses the segmentation technique, noise elimination, k-curvature, and canny edge detector algorithms. As a feature vector, present the palm center's absolute and relative position to define the trajectory and present the contour of the detected fingertips, with these values define the shape of the hand. The classifiers SVM and LIBSVM receive the feature vector. For training, use 400 gestures and 400 to test the model. It has an accuracy of 92.75%, and after applying the blurry technique, it reports 95%. However, it does not report any protocol. It reports a comparative processing time with HMM of 0.95 ms, HOG + DTW of 4.2 sg, and the proposed method 0.43 ms. However, it does not report the protocol for measuring processing times.

In Ref. [41] presents a dynamic hand gesture recognition system using two deep learning techniques and two types of information; deep infrared images and skeleton information. The system recognizes 14 dynamic gestures. To test the model, they used a public dataset called DHG-14/28 and built a dataset with the Intel RealSence F200 sensor. The dataset presents 2800 samples, 1400 images of 640×480 depth of 16 bits, and 1400 samples of 22 joints in 2D and 3D. Also, they used 20 people, each one executing 5 times the gesture. The model gives the classifier data of the standardized images [0–1], and the x, y, and z positions of the skeleton. The deep learning techniques are convolutional neural networks (CNN) and recurrent neural networks (RNN). The model is trained using 1300 depth images and 1300 information samples from the skeleton. To test the model, they use 70 images of depth and 70 samples of information from the skeleton. It reports an 85.46% excavation with 14 gestures and 74.19% testing with 28 gestures.

Reference [42] proposes a new gesture recognition method based on invariant wavelet moments and distance values to improve similar gestures' recognition rate. The model proposes four modules: data acquisition, preprocessing, feature detection, and classification. The system recognizes 10 static gestures. The input data of the system are depth images captured by a Kinect sensor. The dataset consists of 3000 samples, constructed with six people. In the preprocessing, they present the image segmentation by separating the hand region. As a feature vector, it presents the distance from the fingers to the segmented region's centroid. Also, it presents the invariant moments of the wavelets of the hand region. The wavelet transform is combined with the invariant motions, which is the radial component of the kernel transform and replaced by the wavelet basis. The SVM classifier receives this feature vector. The model is trained with 2500 samples and tested with 500 samples. They do not report the processing time but report that they achieve 99.6% classification accuracy. In the paper, they explain that they used data from the same person for training and testing. However, in the design of another experiment where they use data from other people for training and others for testing, they present lower values.

In Ref. [43], a virtual reality system based on hand gesture recognition is presented. The system recognizes four static gestures, and the LMC captures the data. The system uses the spatial positions and orientation of the fingers and palm. The authors do not evidence the construction of a dataset. Using normalized data as preprocessing, the authors normalize the data by dividing each of the fingers' distance by the middle finger's maximum distance by sizing values between [0, 1]. They mention that the selection of features that best represent the gesture and its size is most important for classification accuracy and processing time. In this sense, as a feature vector, they use the distance between the fingertips and the center of the palm. Also, they gather the fingertip orientation vectors and feed them to the classifier. The classifier used is kNN with Manhattan distance and a $k = 3$. They report a classification accuracy of 82.50% and a processing time of 0.057 ms in recognizing a gesture. It does not report how much data they trained the model on. They do not report a protocol for measuring processing time.

In Ref. [44] presents a novel framework of classification and recognition of static, dynamic, and sequence gestures. The framework is composed of three layers. Layer one presents the classification of motion components, layer two presents the classification of location components, and layer three presents the classification of shape components. To validate the system, they use the ChaLearn Gesture dataset. This dataset, designed for one-shot learning, consists of 50,000 samples, composed of 500 batches, each containing 47 sequences, and each sequence 1 to 5 gestures. Each sequence draws 30 gestures. To build the dataset, 20 people participate. The Kinect sensor acquires the data, and the data are 240×320 depth images with a sampling frequency of 10 frames/s. In this paper, it is difficult to determine the proposed generic architecture. However, in each layer, they present different techniques, such as principal component analysis. In another layer, they present a weighted dynamic time warping WDTW technique. In another layer, they present the classification with the one-shot learning algorithm.

In the paper, they present results of processing times by layers, but the result of the three layers is 3.75 sg. This system cannot denominate in real-time because it does not work in less than 300 ms. In the same sense, they present the results of recognition by layers and compare the results with traditional techniques such as HMM-DTW.

In Ref. [45] presents a novel technique for sensing the hand. It uses a technique based on the sensor's proactive movement, which avoids the occlusion of the fingers. To validate the proposal, they classify 1 static gesture. The device used is the LMC. It built a dataset using 17 people and obtained 2000 samples, with a sampling frequency of 60 frames/sg. The captured data are the spatial positions, speed, and direction of the fingers and hand. In the pre-processing, they use the technique of dimensional reduction. It uses the type of hand as a feature vector, if it is left or right, the swingarm angle, the palm position, its speed, and the orientation. The classifier used is kNN algorithm, and they report 93% of classification accuracy. They do not report protocols but mention that the system improves hand estimation when capturing data, and therefore does not require a ground truth of the gesture.

In Ref. [46] presents a new hand gesture recognition system based on the shape and stroke similarity, and mention that solve the ambiguity of the traditional recognition systems based on templates, especially with non-conventional and very similar gestures. They use three databases to train and validate the system. The first one is called EDS1, and it is formed by 5040 samples of 14 subjects with an average age of 21 years. It does not mention the device with which they acquired the data. The second one is called Unistroke, and it contains 1600 samples, it is built with 10 subjects, and it does not mention the age of the subjects, neither the device with which they acquired the data. The third one is called Authentication, which consists of 624 samples built with the Kinect device, built with 13 subjects. In the pre-processing module, they use a media filter to remove jitter. They also remove noise and eliminate the gesture's initial and final points because they start and end with a stationary period. Finally, they normalize the data, and they take as a feature vector the segmentation of the images taken by segments to what they call sub-strokes. Finally, they compare the segments instead of comparing the whole gesture using DTW. They do not mention the amount of data they used to train and test the model, but they report a classification accuracy of 90.05% and variability of 5% with the first dataset. With the second dataset, they report 98.20% and 5% variability, and with the third dataset, they report 97.80% accuracy and 15% variability.

In Ref. [47] presents a hand position recognition and tracking system based on a deep neural network. The authors present this system due to the limitation and complexity of extracting a feature vector in a manual way that adequately represents the gesture. The system recognizes 36 classes between static and dynamic that represent the American Sign Language. To train and validate the system, they use a public dataset called LeNet-5, and it is composed of in-depth data acquired by the Kinect. This dataset contains 507,000 samples acquired with a sampling frequency of 30 frames/s. As a pre-processing, it presents the image segmentation. The model uses a deep neural network; the network is trained with 338,000 samples and tested with 169,000 samples. It reports 98.12% classification accuracy and processing time of 0.899 ms. They compare the obtained data with other models, and its data reports higher accuracy and shorter processing time.

In Ref. [48] proposes a new distance metric called canonical superpixel-graph earth mover's distance. To evaluate this proposal, it uses five public datasets. These are: hand digital dataset NTU, hand gesture dataset HKU, multi-angle hand gesture dataset, this is an extension of the dataset HKU. These datasets are built using the Kinect and have 1000, 1000, and 3000 samples, respectively. The fourth dataset is the Microsoft Kinect and Leap Motion Controller Dataset, and it contains 10 different classes, each repeated 10 times by 14 different people, the fifth is the creativeSenz3D dataset, it contains 11 classes, each class is repeated 30 times by four different people, it presents 1320 samples. As preprocessing, they present the normalization and alignment of the hand shape according to the palm's depth and centroid values. The classifier used is kNN with the proposed distance metric. To train and validate the model, they use the cross-validation technique. The authors do not present processing times but present values of classification accuracy for each dataset described above. These are 99.70, 99.40, 97.90, 96.60, and 97.40, respectively. It does not present a protocol of how to report data.

In Ref. [49] presents a hand gesture recognition model based on a CNN and an LSTM-type convolutional neural network. The authors mention that they use the combination of two types of convolutional neural networks because CNN discovers correlation patterns between the images. At the same time, LSTM specializes in discovering patterns in time frames related to skeletal movements. The model uses five public databases; MSR action 3D presents 557 samples and data acquisition protocol, MSRDailyActivity presents 320 samples, UTKinect-Action3D presents 199 samples. Moreover, ten different people built each one of these datasets using the Kinect device. Also, they use the NTU-RGB + D dataset, which presents 56,880 samples without mentioning the device but contains 3D infrared data of the skeleton. The Montalbano dataset presents 14,000 samples without mentioning the sensor used, the type of data acquired, nor the people that built it. The DHG 14/28 dataset presents 2800 samples acquired using the Intel RealSence F200 sensor. This sensor receives the skeleton's information with a

sampling frequency of 30 frames/s, and 20 people built this dataset. In the paper, they report recognition accuracy values for each dataset these are: 96%, 63.1%, 99.0%, 67.5%, 79.15%. The mentioned demonstrates that the accuracy of the model depends on the type of gesture.

In Ref. [50] presents a hand gesture recognition model, and to validate the model presents an application. The model recognizes five classes. The LMC acquires the data; the system's data is the spatial data from the fingertips and the palm. It also uses the orientation of the hand. The authors present a proprietary dataset built with 10 subjects and composed of 5000 samples captured at a frequency of 60 frames/sg. As pre-processing, they used a mean filter with the window splitting technique with $w = 25$. As a feature vector, it presents the distance between the fingertips and the palm center, length of the fingers base, the thickness of the fingers, and the distance between the fingertips of adjacent fingers. Also, it presents the angles between the fingertips and the palm center, and the between fingers. The model tests two classifiers, SVM with radial base functions. And Naïve Bayes. In both cases, it uses cross-validation to train the model and reports 99.58% classification accuracy with 6% variability and 98.74% with 3.64% variability.

In Ref. [51] presents a novel multimodal framework by sign language recognition using two devices, the Kinect, and LMC, which also use two classifiers algorithms. This system recognizes 50 dynamic sign-words; the words correspond to Indian Sign Language. For training and testing the framework, they build a dataset with 7500 samples using 10 people, eight males and two females. When using the Kinect device, the sample's frequency is 30 frames/sg, while when using the LMC, the frequency of the sample is 100 frames/sg. The data of the two devices it acquired at the same time. The data acquired by the devices were fingers and palm positions. As the sample frequency is different in used devices, the authors used techniques for pre-processing such as resampling, correlation, and software synchronization also normalized the sign to $-1, 1$. Feature extraction presents 11 features for static gestures and 22 for dynamic gestures based on fingertips positions, palm center, and fingertip direction. This data fed two classifiers HMM and bidirectional LSTM-NN (BLSTM-NN). For training and validate the system, the authors use 6000 samples and cross-validation techniques, and for testing, use 1500 samples. They reported 97.85% and 97.55% as the accuracy value of recognize. Do not report time of processing, nor protocols of measure of accuracy.

In Ref. [52], a human hand pose retrieval system is presented. This paper was considered for the systematic literature review because it retrieves data from the dataset and, through supervised learning methods, labels these hand poses or part of the hands. The authors retrieve 2D images and represent them in 3D poses. For this process, they used autoencoders neural network. For this process, they use two public datasets, MSRA and ICVL. These datasets were constructed with an interactive gesture camera device. This device captures depth images. 6 subjects built the MSRA dataset and present 2400 samples, while the ICVL dataset was built with 10 subjects and presents 180,000 samples. To train the model that recognizes hand poses, they used an artificial neural network (ANN) with two hidden layers and used the RELU function as the activation function. The ANN was trained with 178,600 samples and tested with 1400 samples. They do not report accuracy. The authors present the problem as a domain adaptation and treat it as a semi-supervised learning problem, where the training set is labeled, and the test set is unlabeled. In this sense, employing supervised learning, they recover the label that corresponds to the test sample.

In Ref. [53] presents two hand gesture recognition models. The first model recognizes static gestures, and the second model recognizes dynamic gestures. The authors present three modules: detection, tracking, and recognition. To evaluate the models, they built a dataset called UESTC-HTG. It includes 1600 samples, and they used 100 subjects. Besides, it presents a second dataset called UESTC-ASL with 1100 samples, and they used 11 subjects. For capturing the data, they used the Kinect device that returned a 3D hand trajectory. For evaluating this paper in our proposed architecture, we present the techniques used in pre-processing: binary images, extracting contour with canny edge algorithm, filtering gesture trajectories, and decomposition strokes. They classifier the gestures using DTW, specifically comparing warping of functions that represent input images and saved classes. They reported 98.44% accuracy using the UESTC-HTG dataset. This value is the average between all runs.

In Ref. [54] presents two models of hand gesture recognition. The first model recognizes nine static gestures, the numbers between 1 and 9. The second model recognizes six types of dynamic gestures. However, it does not mention the device used to acquire the data. The authors do not mention the dataset's construction but mention working with 750 samples and 13 subjects. They are pre-processing the images for eliminating noise and also segment the regions of interest for recognition. To generate the feature vector, they use HOG of the interest region segmented and propose extracting nine weighted moment features from the smoothed images. These features fed to the classifiers. For the classification of static gestures, they use SVM, while for the classification of dynamic gestures, they use HMM. To train the model, they use the cross-validation technique and report a processing time of 0.04 s. Besides, it reports 93.78% classification accuracy for static gestures, and it presents 93.5% classification accuracy for dynamic gestures. It does not report measurement protocols of the reported values.

In Ref. [55] presents a gesture recognition system based on depth images and a deep neural network. To validate

the model, the authors present an application based on 41 signs of Japanese Finger-spelled Sign Language. They use the Kinect sensor to acquires the data. The authors present their dataset with 3280 samples, built with eight subjects, five men and three women. In the pre-processing, they segmented the hand, wrist, and hand palm regions of the images using the Time-series curve technique. This data fed to a deep neural network. This deep neural network contains three convolutional layers, three max-pooling layers, two local response normalization layer, three RELU activation layer, one inner product layer, and one SoftMax loss layer. To train the model, they use 2870 samples and the cross-validation technique. To test the model, they use 410 samples. They report a processing time of 2.3 ms. With this model, they have an accuracy of 88% and compare with a previous model of the same authors who used HOG + SVM and reported 84%.

In Ref. [56], a model of recognizing three dynamic gestures applied to a human–computer system that allows moving a menu is presented. The data used for the system are the spatial positions of the hands in 3D captured by the LMC. The authors present a data set with 100 samples. However, they do not mention how many people built the data set. Additionally, they mention that in the proposed system, the time taken to open and close the hand is 2 s, and in the 2 s, a sample of about 140 frames is acquired. However, these time series were sub-sampled to 10, 20, and 30 frames, and these time series are delivered to the classifier, respectively. The classifier is a recurrent neural network, but they do not mention the architecture of the network. To reduce the range of gesture variations and increase the classification accuracy, the authors transform the mentioned spatial time series into a time series describing the vector differences. This time series was produced by recording increments between two successive coordinates, which were then normalized as unit vectors. In this sense, they present a classification precision of 80.00% with a variability of 0.1. They mention that the system works well for simple gestures, while for complex gestures, the accuracy rate is low, they do not describe values. Additionally, because of the small amount of data, it is assumed that overfitting can occur.

In Ref. [57] proposes an American Sign Language recognition system based on depth imaging and a convolutional neural network. They mention that the problem is very complex because of the similarity of the fingers' gestures and the occlusion. They propose to recognize 24 signs and 10 numbers. The data obtained for the development of the work are depth images captured by the Kinect sensor. The authors propose a new system of data increases called multiview augmentation strategy, which can introduce variations of the image in different perspectives through a new number of points obtained from the original image. They build a patented data set of 12,000 samples with five subjects. Also, they use a public data set called NTU created with 10 users. The authors present two modules about the CNN architecture, a feature extraction module, and a classification module. The feature extraction module performs an image size scaling to 32×32 and then normalizes it to a [0 1] interval. They also mention that they use a bandpass filter. These data are sent to a convolutional neural network with three 5×5 layers, using as activation function a RELU that is applied to each convolution. A 2×2 max-pooling layer follows each layer. Finally, the classification module has an input of $4 \times 4 \times 128$. In this sense, 2048 characteristics in two fully connected layers. The authors present an evaluation metric of accuracy, precision, recall, and F-score. In this sense, they present accuracy of 84.8%, an accuracy of 88.9%.

In Ref. [58] presents a real-time static hand gesture recognition system. This system interacts with an application called CAVE. The system recognizes five static gestures, and the Kinect sensor captures the data. These data are depth images and infrared data. In the paper, the authors present a data acquisition protocol and mention that 15 adult persons built the dataset, and the sampling frequency is 30 frames/s. As pre-processing, they use a segmentation algorithm, mean filtering, and the equalization of the histogram. As a feature vector, they use a technique called SIFT (bag-of-visual -words). As a classifier, they use SVM with radial based functions. To train the model, they use 6841 samples, and to test, they use 21,336 samples. The authors present a processing time of 90.2 ms and average recognition accuracy of 89.53% and report the grading accuracy for each gesture.

In Ref. [59] presents a classification model bringing together two types of signals, depth images acquired with the Kinect sensor and 3D data acquired by the LMC. Besides, they present a novel feature vector. The system recognizes 10 static gestures, and the gestures are the numbers from 1 to 10. The authors work with a public dataset, which consists of 1400 samples, and built by 14 people. The images are segmented before extracting the features, while the LMC passes the raw data to the features extractor and extracts the feature of a separate mode. The number of detected fingers, the fingertips position, the palm position, and the finger orientation formed the LMC feature vector. While from the images are: the radius of the hand, characteristics of the curvature, the distance between the fingertips and the center of the palm, correlation characteristics. Finally, these characteristics are put together and delivered to two different classifiers. These classifiers are SVM and random forest. To train the model, they use the cross-validation technique using 1300 samples, and to test the model, they use 100 samples. They do not report processing time, but they report 81.5% of accuracy with the LMC and 96.3% of accuracy with the Kinect. Besides, they present the accuracy values of combination features.

Reference [60] presents a new hand gesture recognition model. It recognizes 10 static and 26 dynamics gestures. The static gestures are the numbers 1 to 10, while the dynamic gestures correspond to American Sign Language. The authors present a dataset composed of 3600 samples, 1300 corresponds to American Sign Language, and 500 samples correspond to the numbers 0 to 9. The LMC acquired the data, and 10 subjects built the dataset. They do not present a data acquisition protocol, nor do they present the sampling frequency. The data used to build the model are the positions of the fingertips and the fingers' orientation. The authors do not present a pre-processing of the signal. Four different characteristics are extracted from the raw data to form the feature vector, which formed from the x and y coordinates of the fingertips, and the angles taken from the direction of the fingertips. This feature vector feeds a neural network with radial-based functions. The model is trained using 1800 samples and tested using 1800 samples, reporting a 94.2% recognition accuracy for static gestures, while for dynamic gestures, it reports 89.2%. They also train and validate the model with 2600 samples using the tenfold cross-validation technique and test using 1000 samples, reporting a validation accuracy of 95.1% for the static gestures, while for the dynamic gestures, they report 92.9%.

In Ref. [60] presents a hand gesture recognition system using virtual reality and using the fusion of two types of signals. The LMC and Myo-Armband acquire these signals. The system classifies six static gestures. They present a dataset with 36,000 samples, built with 11 people, eight males, and three females, each gesture is acquired in 3 s, and 300 repetitions for each gesture. The authors present a protocol of the experiment. In the pre-processing module, the authors present signals interpolation at 100 Hz, due to the difference in the acquired data's sampling frequency. The LMC has a variation of between 30 and 60 Hz, while the FMG has a constant frequency of 50 Hz. Also, the signals were passed through a median filter using the window splitting technique $w = 11$. The angle formed between the fingertips direction and the palm direction plane, the palm center's fingertip distance, the distance between the finger and its tip projected in the palm plane formed the feature vector. They also mention that the normal vector and the palm center are returned by the LMC form the palm's plane. They present four classification algorithms, linear discriminant analysis (LDA), SVM, bagging of decision trees, and neural networks. They present values of classification accuracy, LDA present 85.4% with the variability of 5.5%, the three-bagger report 86.4%, and variability of 5.5%, with SVM 88.3% and variability of 4.7% and with NN 87.5% and variability of 5.6%. However, they do not mention the method used to measure and report these values.

Reference [61] presents a hand gesture recognition system based on hand tracking using Intel RealSense and LMC devices. The system recognizes 26 American Sign Language gestures. The authors mention that some systems work with large feature sets. However, they propose and present a feature set of 30 data, and that with this data vector, they achieve high values of recognition accuracy. The proposed feature set is based on the x, y, and z positions of each finger and the direction values of the fingertips returned by the LMC. However, they mention that they worked with 50 people and developed 250 observations. All the gestures were made with the right hand. They mention that the data were collected one after the other and delivered to the SVM multi-class classifier, built using a Gaussian kernel using the Scikit-learn library and the LIBSVM. The authors mention that the main contribution is to obtain a better classification value with a lower number of characteristics. However, the authors do not mention values for the accuracy of the classification. Neither do they present a pre-processing of the data nor do they present processing to extract characteristics from the presented data vector.

In Ref. [62] mentions that accuracy in hand gesture recognition systems is affected by problems generated in the collection of information, especially by noise. In this sense, the authors present a hand gesture recognition model based on nested interval unscented Kalman filter with the LSTM (NIUKF-LSTM) network, which applied in the dataset with noise. The authors present a dataset formed by 2800 sequences. 28 people built the dataset, and each person repeats between 1 and 10 times each gesture. The paper proposes to evaluate the system using 14 and 28 gestures. They evaluated the model in three parts. First, they evaluate fine gestures, second coarse gestures, and finally, the two types of gestures. Besides, they use four classify algorithms. The first method evaluates an LTSM network, second a UKF-LTSM network, third an average-LSTM, and finally, the proposed NIUKF-LSTM method. The paper reports an accuracy value of 87.82% using NIUKF-LSTM evaluated on 14 gestures, while evaluated on 28 gestures, it has 80.44% accuracy. The paper presents a comparison of results and an analysis. However, it does not mention how they evaluated and how they reported the values, nor does it present a processing time.

In Ref. [63] proposes a gesture recognition system for Taiwanese sign language using the Kinect sensor. The system evaluates 12 directions and 24 forms associated with Taiwanese sign language. The document mentions a data set of 400 samples built with five subjects. The system's relevant information consists of depth images, 3D positions of the hip, wrist, spine, and shoulders. The authors use this data and PCA to extract features such as palm segmentation or hand shape. To segment, they apply the techniques of vertical projection and Otsu binary Threshold. Besides, they use HMM to determine the hand's trajectory because the hand gesture involves a variation of its shape concerning movement time. These are determined by the distance of the

movement, the direction, and the angle. Besides, the distance between the hands, the direction vector, and the angles determine the trajectory. They also extracted the hand's shape using PCA and delivered all the information to the SVM classifier. Besides, they report an average rating accuracy of 89.67%. The article does not mention processing times, data collection, or algorithm evaluation protocols.

In Ref. [64], the authors present a new LSTM network architecture called hybrid bidirectional- unidirectional LSTM (HBU-LSTM) to evaluate dynamic gestures. This architecture is based on the evaluation of the unidirectional, bi-directional, and deep LSTM approach. The proposed method consists of analyzing the temporal spaces during the forward and backward sending in the LSTM network layers, obtaining important characteristics that define very well the gestures involved in the problem. To evaluate the proposal, the authors execute it on two public datasets, the LeapGestureDB and the RIT. The same authors release the first of these datasets in a previously published article. The LeapGestureDB consists of 11 types of gestures, while the RIT dataset consists of 12 gestures. The data of these datasets are acquired by the Leap motion conroller sensor. The LeapGestureDB was generated by 120 volunteers obtaining 6600 samples, with a sampling frequency of 115 fps. In this sense, the authors with the proposed architecture present a classification accuracy of 89.98% on the LeapGestureDB dataset and 73.95% on the 73.95% dataset. To validate that the proposal is valid, they compare the results with the ULSTM, Bi-LSTM, D-LSTM architectures, reporting accuracy of 76.23%, 83.09%, and 86.19%, respectively, these architectures evaluated on the LeapGestureDB dataset. The authors mention that the reported accuracy values are because the gestures are very similar to each other. They also mention that the model does not work in real-time, and neither do they present a protocol for assessing the accuracy of classification.

In Ref. [65], a new and efficient way to extract characteristics from the hand movement is presented. This process is based on the chronological indexation of time patterns identified in a time series, which presents a sequence of hand movements called gestures. The data acquired for the work is done with the Leap Motion Controls. However, the paper's authors describe device orientation problems, lighting, hand size, and finger occlusion. For this reason, they present a data acquisition protocol, despite working with the LeapGestureDB dataset generated in a previous study by the same authors. As a pre-processing, they use the wavelet transform (Daubechies wavelet filter) of order 2. They mention that it is an effective method to denoising raw data generated by LMC. The characteristic vector is also evaluated with the RIT dataset. Once the data has been pre-processed, and the new data vector has been formed, it is fed into the classifier. As a classifier, they use KNN with a Hamming distance. Besides, the authors divide the dataset into five groups to use the cross-validation technique. In this context, the authors report the accuracy values of the proposed LeapGestureDB and RIT datasets of 95.45% and 93.38%, respectively. However, the authors do not present a measurement protocol. They also mention that the times obtained do not consider the work in real-time. To validate the proposal's effectiveness, they compare with other algorithms such as DTW, HMM, LSTM, GMM and report accuracy values of 89.4%, 52.23%, 76.23%, 51.36% on the LeapGestureDB dataset.

In Ref. [66], the authors mention that there are no works that recognize static and dynamic hand gestures simultaneously. In this sense, they propose a model for the recognition of hybrid hand gestures. The model uses three modules: data acquisition, pre-processing, and classification. For data acquisition, they use the Leap Motion Controller and form a dataset of 600 samples. Each gesture is repeated 30 times, and the dataset is formed by 10 static gestures and 10 dynamic gestures. As a pre-processing technique, they present the resampling of the data and the transfer of the coordinated axis from the center of the Leap Motion to the center of the palm. And as a classifier, they use CNN + LSTM. The authors also present a technique to increase the dataset's size because, with very little data, the model could generate an overfit. The technique used is resampling using the Fast Fourier Transform. In the paper, the authors refer to real-time work, but in the study, they do not refer to the model's response time but rather to the way the data is collected. In classifying the gestures, the authors use the cross over technique with a k-fold of 10, presenting an excavation value of 98.83%. The authors compare the result obtained with other classifiers such as SVM, DTW, and HAR-CNN, presenting 83.33%, 87.68%, and 98.00%, respectively. However, the authors do not present a measurement protocol to report the obtained values.

In Ref. [67], a human–computer interaction system based on hand gesture recognition for Arabic sign language is presented. The system presents a model composed of six phases, data acquisition, normalization, feature extraction, gesture classification, Ada-Boosting implementation, and recognition. The system recognizes 42 signs between static and dynamic. The authors for the data acquisition use the Kinect device, capture spatial positions such as hand, shoulder, elbow, middle spine, and the palm center. They build a dataset of 1260 samples. Besides, they normalize the data because the size of the persons is variable. Also, they form a coordinated axis taking as an axis the data of the middle spine. To form the feature vector, the captured data are transformed to spherical coordinates and calculate the radial distance r between the mean spine's data and the other points. Also, they calculate the angle between the positive axis of the x-axis and the distance r. Also, they calculate the angle between the positive axis z and the line formed between the

origin and the point. This feature vector feeds two classifiers to Random Forest and Naïve Bayes, these classifiers return a label vector, and most votes select the label returned by the classifier. To improve the sorting system, they use the Ada-Boosting sorting assembly technique. The authors take as recognize the value of the label returned by the sorter assembly process. To train the model, the authors collect 20 samples of each of the signs, while for testing, they collect 10 samples of each of the signs. For training, they use 840 samples, and for testing 420 samples. The authors report the accuracy values of 91.18% for Random Forest, 92.50% for Naïve Bayes, and 93.7% applying Ada-Boosting. Besides, they compare the results with DTW and HMM, reporting 77.8% and 79.5%, respectively. The authors also report execution times: for Random Forest 15.3 s, while for Naïve Bayes 8.09 s.

In Ref. [68], the authors present a UAV flight control system based on hand gesture recognition and deep learning. They mention that in the field of the application domain is the first article published. The system consists of three subsystems. The first consists of data acquisition, the second is to pre-process the data, extract a feature vector, and classify the gesture, and the third is to control the UAV based on the gesture returned. However, our interest is focused on the two first subsystems. In this sense, the data acquisition subsystem presents a dataset with 11,000 samples, created from data capture with the Leap Motion Controller. They propose to recognize 10 dynamic gestures. Each gesture is composed of a 45×15 matrix, where 45 corresponds to the captured frames and 15 to the raw data that the device delivers. Besides, the authors mention that the more distant the range of independence of the characteristics, the better the classification results. In this context, they perform data normalization and work with two datasets, one normalized and the other with the raw data. These two datasets will feed three different types of neural networks for training and testing individually. The first neural network with two layers, the second with five layers, and the third an 8-layer convolutional neural network, the latter presents three convolution layers, giving the last layer 200 neurons for classification. The subsystem is trained with 9124 data and tested with 1938 data. They present accuracy values for the three network types and for each proposed network architecture, where it is observed that the highest digging returns with the normalized dataset and with the five-layer network corresponding to 98.555%, however, it is also observed that the convolutional neuronal network behaves well with the raw data dataset with an accuracy of 97.626%.

In Ref. [69], they present a system for controlling CAD interfaces. This article does not meet the exclusion criterion that corresponds to remove the articles that present only applications because it presents a preprocessing of the data and an interesting feature vector. For data acquisition, they use the Leap Motion Controller and form a dataset of 480 samples with a sampling frequency of 60 fps. The dataset is formed by 30 volunteers who repeat each gesture twice. The model recognizes eight dynamic gestures, which are clearly explained in a video. As a pre-processing, the authors present the calculation to move the coordinate axis from the Leap Motion to the palm's center. Besides, these last data are transformed, employing a calculation with a rotation matrix. From this data, they obtain the feature vector. This vector is composed of the angles between the direction of the arm and the palm's direction, the direction of the hand, and each of the phalanges that compose the fingers. This calculation for each of the fingers, understanding that the thumb only consists of two phalanges. The concatenation of all calculated angles forms the feature vector that will feed two recurrent neural networks, such as LSTM and GRU. They mention that these two types of networks are often comparable in terms of accuracy. The two networks receive 18 features as input and are connected to two hidden layers, each composed of 200 LSTM neurons. Also, as the dataset is tiny, the authors present a data augmentation technique, which is not well explained. However, for the training and testing process, they present three datasets TSbase, TSA1, and TSA2, on which they report accuracy values. For LSTM they report 87.3%, 91.6%, and 93.7 for each one of the datasets, in the same sense for GRU 84.3%, 87.5%, 88.5%. The results obtained are compared with SVM on the same datasets and report 70.8%, 75.0%, and 71.8%, respectively. In this context, the authors comment on the importance of having a significant dataset.

3.3.1 RQ1: What is the architecture of the proposed models for hand gesture recognition based on machine learning and infrared information?

The gesture recognition means that a machine knows or interprets a gesture developed of a human with her hand, face, or body. In the same sense, a study mentions that 21% of people use the hand for communication with a machine or other people. For recognizing these gestures using machine learning, the researchers propose different models. The models are a set of modules that contributes to hand gesture recognizing without using the mathematical models [13].

The variable that contributes to answering this research question is the *model structure* described in section 2.3.3. It is composed of data acquisition, pre-processing, feature extraction, classifier, and post-processing modules. It is necessary to mention that the architecture of a machine learning model is composed of the union of all or some of these modules. In this context, the form of retrieving information for evaluating this variable is mapping the architecture proposes with the architecture of models found in the scientific literature. In the sense of showing how many and which

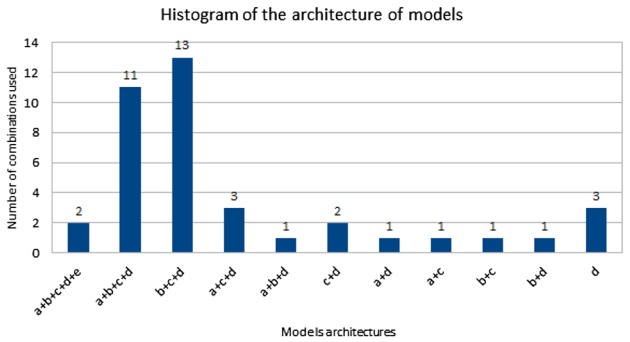

Fig. 1 The architecture of the model proposed vs. architecture of models found in the scientific literature; The parameters of the model structure proposed are: **a** data acquisition, **b** pre-processing, **c** feature extraction, **d** classifier, and **e** post-processing

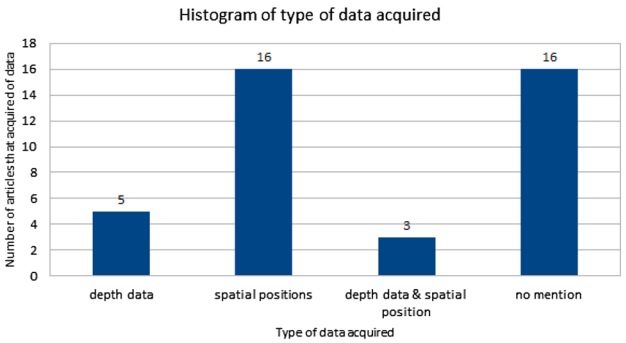

Fig. 2 Type of data used for intending resolve the problem of hand gesture recognition using machine learning and infrared information

modules are used by the authors of the papers in the presented models.

Figure 1 presents a histogram representing a correspondence between the number of the modules combinations offered by the authors and the number of papers that use these combinations. For explanation purposes, since it is not convenient to write the name of the module, it is abbreviated as (a) data acquisition, (b) pre-processing, (c) feature extraction, (d) classifier, and (e) post-processing. So, the combination of these letters expresses a model found in the scientific literature.

According to the architecture proposed, most papers used the pre-processing, feature extraction, and classifier modules. However, this architecture adds the data acquisition module. This module used in all models presented. However, not all works describe the form and the amount of data acquired.

It is necessary to describe that in each of the modules presented, the researchers use different techniques to achieve maximum classification accuracy and are detailed in the following paragraphs.

In this context, in the *data acquisition module*, the systematic literature review reported that for hand gesture recognition using machine learning and infrared information, there are two types of data with which the problem can be addressed: spatial positions and depth images. The spatial positions are relative in a device that can recognize objects. This type of data is the most widely used to address hand gesture recognition using infrared information.

Figure 2 shows the number of articles that address the problem using both spatial positions and depth imaging. Also presented are papers that use both types of data simultaneously.

The second type of data most used is depth images. These types of images contain the distance between objects and devices that contain the depth cameras. These types of images typically are used for forming 3Dimages. The scientific literature also presented articles that use both types of data, special position, and depth data. In the same sense, many articles do not mention what process used for data acquisition.

For data acquisition, the systematic literature review reported the use of the following devices: Leap Motion Controller, Kinect, Intel RealSence, and Interactive Gesture Camera. However, the devices most used are the Leap Motion Controller and Kinect. The Leap Motion has three led sensors and two depth cameras. This device is specialized for tracking hands and fingers. Frequently for the data acquisition module, its sample frequency is 200 frames by second. Themselves, the Kinect is a device imprecise with hand and finger tracking; its sample frequency is 30 frames by second. These devices for interacting with a PC use their SDK.

The *preprocessing module* consists of transforming an input signal into another signal. The resulting signal can be a noise-free or normalized. This increases the possibility for the classifier to obtain better results. During the systematic literature review for this module, it is observed that researchers use different techniques to transform the signal. The techniques reported in this module are dimensionality reduction, normalization, noise reduction filters, manual segmentation, image equalization, movement of the sensor coordinate axis to the center of the palm, and articles that did not report the technique.

Figure 3 shows the relationship between the techniques used and the number of papers using these techniques. Many of the papers use more than one preprocessing technique simultaneously. As the names are long to express in the graph, they are symbolized by (a) dimensionality reduction, (b) normalization, (c) filters to reduce noise, (d) segmentation, (e) equalization, (f) no mention, (g) movement of the data concerning the palm.

The papers that use dimensionality reduction techniques, they used principal component analysis, and autoencoder type neural networks. Also, the papers that used the

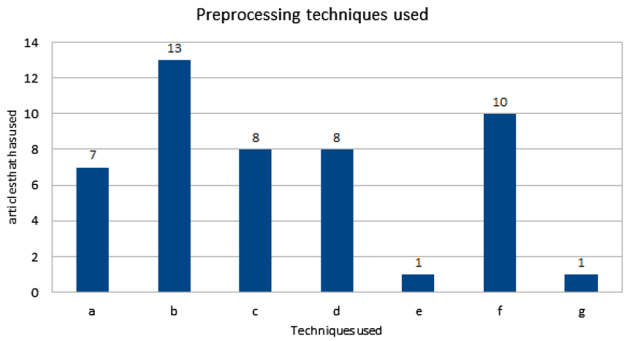

Fig. 3 Techniques used for pre-processing vs. the number of articles that used these techniques

normalization techniques used the values between 0 and 1. For noise reduction, the papers mentioned that used the Kalman filter, discrete wavelets transform, and the median filter when the data was images. Finally, papers that used segmentation techniques used canny filter and k-curvature, and manual segmentation.

The *feature extraction* is one of the most important modules for obtaining a higher accuracy of classification or recognized [70]. Feature extraction consists of obtaining a vector of values that characterizes a gesture sufficiently well so that it can be differentiated from other gestures. We observed that the authors present manual and automatic feature extraction techniques during the systematic literature review. For manual extraction techniques, the authors propose novel and complex functions that require a great computational capacity and knowledge. Also, the authors present automatic feature extraction techniques based on deep learning algorithms.

Figure 4, presenting the techniques that scientific literature presented for feature extraction. The names of the techniques that are too long to represent in the figure are represented with corresponding letters: (a) segmented image, (b) statistical measures of central dispersion, (c) distance and

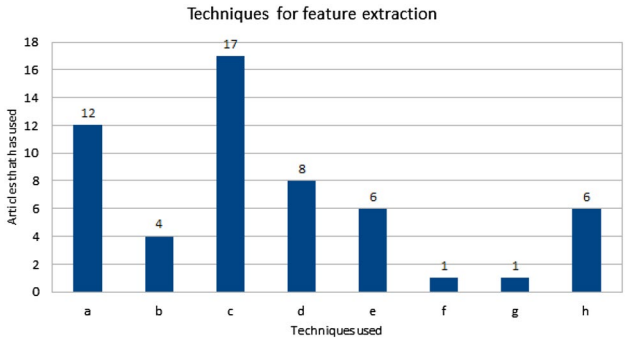

Fig. 4 Techniques used for feature extraction vs. the number of articles that used these techniques

operations of spatial position, (d) angles of spatial position, (e) convolution, (f) HoG, (g) chronological-pattern-indexing, and (h) no mention.

Between the feature extraction method reported in the scientific literature are angles between the spatial position from the fingers, statistical measures of central dispersion, features from segmented images, the distance between center palm and fingertips, mathematical operations from spatial positions, and convolution.

Table 6 shows the techniques used when papers using angles of the spatial positions as a feature vector, also revealed the accuracy values reach. This report shows a value of 99.58, and this accuracy reached using a dataset of 5000 samples.

3.3.2 RQ2: What are the protocols, types of sensors, and types of the DataSet, used for developing hand gesture recognition models based on machine learning and infrared information?

A protocol is a set of rules or instructions that need to follow to attempt to maximize results. In this sense, it is necessary to know if the primary studies present protocols for data acquisition and for presenting results. The data acquisition protocol must clearly describe the gestures to be performed, the hand location in front of the sensor, sampling time, number of users, age, gender, number of repetitions per gesture, and if the people had problems with their hands. Also, it must specify how presented the results of accuracy, classification, and recognition.

At the same time, it is necessary to know the types of sensors that primaries studies reported in contrast to infrared sensors types. Because the methods, techniques, problem, and dataset depend on the type of sensor that is used.

Finally, it is necessary to know if the dataset used is public or private, the number of samples, and the number of subjects with which they constructed.

Initially, we presented the types of sensors reported. These are Leap Motion Controller, Kinect, Intel Real Sence, Interactive Gesture Camera. Do not all works presented reported the types of sensors used, also, in Fig. 5 shown how evolved the use of the sensors for years.

Table 6 Percent of accuracy reported by articles that use angles of the spatial positions

Variables used	% Accuracy reported
Angles between wrist and fingertips	93, 86, 96.41, 97.25, 99.58, 88.3
Angles for each joint (3-angles)	
Angles between each two bones	

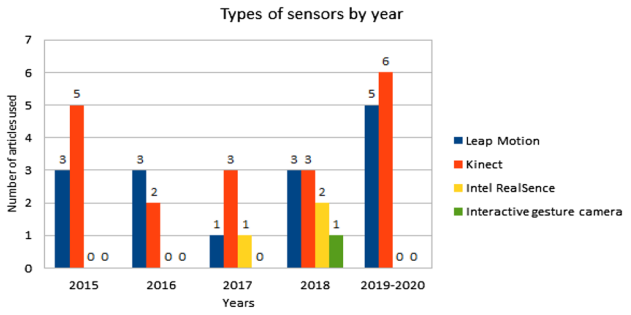

Fig. 5 The types of sensors reported in the scientific literature for attempting to resolve the hand gesture recognition problem using machine learning and infrared information

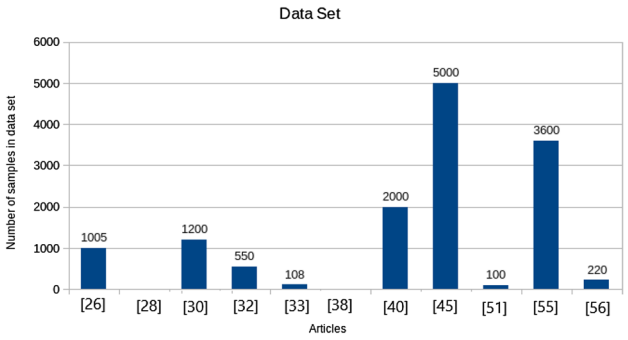

Fig. 7 It reports the number of samples, the datasets built with Leap Motion Controller, and articles that use

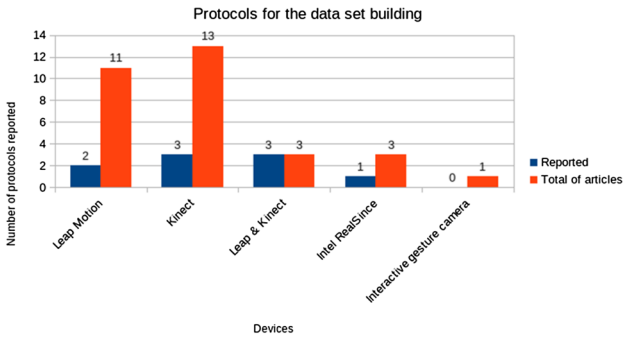

Fig. 6 Articles that report the use of protocols for building datasets

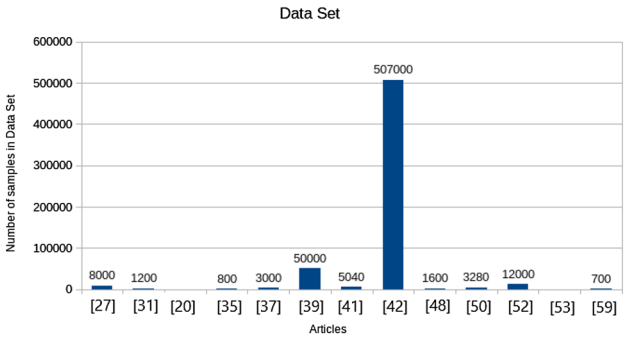

Fig. 8 It reports the number of samples, the datasets built with Kinect, and articles that use

The Kinect is a sensor that samples up to 30 frames per second, has cameras in RGB and infrared, returns 30 spatial positions of the whole human body, in the specific case of the hand only returns a general spatial position, the articles that report works using this device they mostly work with images. While the LMC samples at 200 frames per second, is specialized to work with the hand, it returns the spatial positions of all the hand joints, report the direction vector, hand speed, and a normal vector from the center of the sensor to the palm center. Additionally, it returns images in depth.

Likewise, the review shows that only nine articles report the use of a protocol, which puts the reproducibility of the remaining articles at risk. Generally, the articles that report works with the Kinect work with images, the articles that report works with the Leap Motion work with the spatial positions. However, three articles work with the Kinect and with the Leap Motion simultaneously. All articles that work with both types of sensors report protocols. Figure 6 shows the protocols used for building a dataset for each type of sensor.

Since machine learning algorithms learn from data, it is necessary to review the types of datasets used in primary studies and how constructed. Figure 7 shows the datasets built using the Leap Motion, the dataset with the most significant number of samples is 5000, and this data set is built with 10 people and for five types of gestures. However, it reports a classification accuracy of 99.58% with the variability of 6% with the SVM classifier and 98.74% accuracy with 3.64% variability with the Naive Bayes classifier.

In the same sense, Fig. 8 shows the datasets built using the Kinect. The dataset with the most significant number of samples is 507000, and this data set does not report the number of people, the study reports that it created to work with 36 types of gestures, reports a classification accuracy of 98.12% does not report variability, reports that used 338,000 samples for training and 169,000 for testing.

3.3.3 RQ3: What types of learning (supervised learning, semi-supervised learning, unsupervised learning, or reinforcement learning) have been used to train hand gesture recognition models with infrared information?

Machine learning techniques use types of learning, such as supervised, unsupervised, semi-supervised, reinforcement learning, among others. In this context, the problem of hand

gesture recognition using machine learning and infrared information, reported that all articles used supervised learning to train their models.

In the same sense, we analyzed the type of techniques used to adjust the parameters in the different modules of machine learning models. In the data acquisition module, it presented that the parameter adjustment is performed in a heuristic way by trial and error. In the pre-processing module, it adjusts the parameters using the heuristic trial and error technique. While in the feature extraction module, the parameter setting is done by trial and error, but the parameters can also be adjusted automatically. The algorithms used for classification perform the work automatically. It consists of mapping the input data with their respective labels, and the trained model will be able to return a label for a new data set.

3.3.4 RQ4: What are the processing time and recognition accuracy of hand gesture recognition models that use machine learning and infrared information?

The hand gesture recognition problem's processing time is considered a critical variable due to the complexity and many mathematical calculations in the machine learning models. In this context, obtaining values representing a high classification rate and the high recognition rate is a challenge.

From the articles selected for data extraction, 32 articles report the classification accuracy, and only 10 of these present the processing time.

In this sense, only six articles report a protocol to report the classification values, while of the articles that report processing values, none report a protocol. Table 7 shows these data.

Many models reported accuracy over the hand gesture recognition problem. However, these models are not comparable due to the origin of the dataset, the construction of the dataset, the type and the number of gestures, different techniques used for pre-processing and feature extraction, and finally, the classifiers used. The accuracy and classification speed are reported below with the most used classifiers.

The systematic literature review of hand gesture recognition presents that the SVM algorithm is the most popular for classification. For it uses linear and non-linear classification. In its architecture, use kernels. The kernel takes the input data and transforms it, consists of returned an extended dataset, then the boundaries between its classes are most apparent, and the SVM algorithm can compute a much more optimal hyperplane, as shown in Table 8.

The RNN is a family of classifiers for processing sequential data that use a mechanism for holding memory data in hidden layers. The SLR reports that for classifying dynamic gestures, use RNN. Below presents the article that reports the highest accuracy and processing time with this type of classifier, as shown in Table 8.

The kNN is a non-parametric algorithm. In this sense, it depends on the quantity of data, and this algorithm assumes that exist similar features between data and nearby. The kNN algorithm measure proximity with some distance metrics. Below presents the article that reports the highest accuracy K-NN classifier, in Table 8 shown the values (Fig. 9).

Finally, we present articles that used deep learning, each one with its accuracy, in the same sense that the above paragraphs, we present the article with the highest accuracy with Deep Neural Networks in Table 8 and Fig. 10. It consists of a Neural Network with many hidden layers and many neurons in each hidden layer.

Table 7 Articles that report the use of protocols about accuracy and speed processing

Variables used	Accuracy reported
Articles that report a protocol for accuracy classification	[26, 30, 42–44, 61]
Articles that report a protocol for speed of processing	

4 Experimentation

Re-producing the articles becomes challenging because most of the authors do not release the code or the datasets. In this sense, the maximum effort is made to be able to replicate

Table 8 Reported values by classification algorithms

Algorithm	References	Classes	Accuracy	Dataset	Device	Training	Test	Real-time	Variability
SVM	[45]	5	99.58 98.74	5000 samples	LMC	5000	Cross validation	NI	(±) 6%
SVM	[27]	16	98	8000 images	Kinect	4800	1600	0.25 0.3 s	NI
SVM	[37]	10	99.6	5000 images	Kinect	2500	500	NI	NI
RNN	[26]	6	98.4	1812 samples	LMC	1812	Cross validation	125 ms	NI
KNN	[43]	10	99.7 99.4	1000 deep images	Kinect	1000 LpO	Cross validation	NI	NI
DNN	[42]	36	98.12	507,000 deep images	Kinect	338,000	169,000	15.25 ms	NI

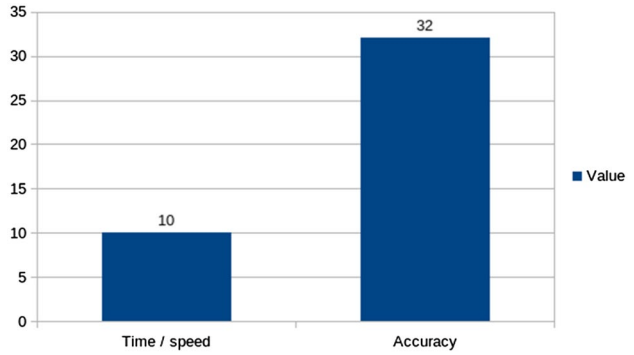

Fig. 9 The articles that report the accuracy and speed of processing

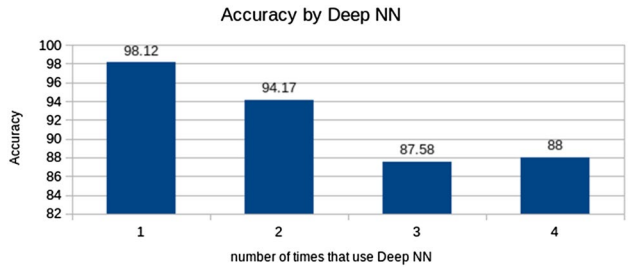

Fig. 10 Articles that report recognition accuracy obtained by Deep Neural Networks

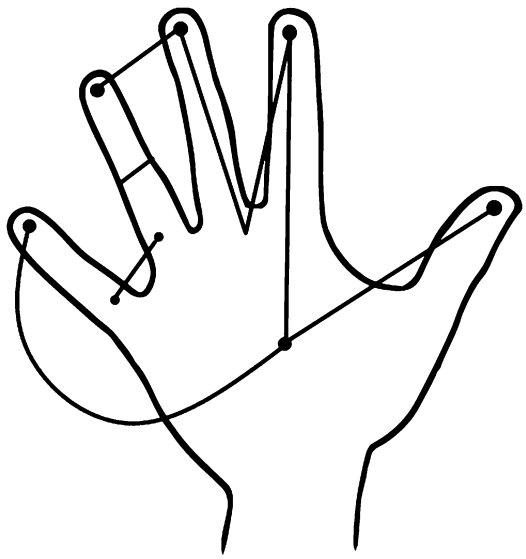

Fig. 11 Represents the spatial positions captured by the LMC that the authors use to construct the feature vector

the proposed architectures. And it becomes evident the need to propose a free standard database so that researchers can test its models.

In Ref. [50] they present an interesting feature vector based on the number of fingers, the distance between two points at the base of the fingers, the thickness of the fingers, the angles between the two closest fingers, the angles between a given finger and the first finger with respect to the palm position, the distance between adjacent fingertips and the distance between the tip of each finger and the palm position, as shown in Fig. 11.

They also present *seven formulas* to calculate features specified. In the *first formula*, the sum of all the spatial positions in x, y, and z of each finger's fingertips is made. The *second formula* calculates the distance between the base of the contiguous fingers. The *third formula* takes the value of the thickness of the fingers. The *fourth formula* calculates the angle between the two nearest fingers. The *fifth formula* represents the angle between a given finger and the first finger with respect to the palm's position. The *sixth formula* refers to the distance between adjacent fingertips. And finally, the *seventh formula* calculates the distance between each of the fingertips and the center of the palm.

The authors do not release the dataset either the code. In this sense, for reproducing this paper, we work in Matlab and reproduce these formulas over a dataset built with 56 users, the same that execute nine gestures. Each gesture is repeated 30 times. The dataset we use does not have the base of the fingers nor the thickness. In this context, we proceed to apply the first, fourth, fifth, sixth, and seventh formulas.

For reproducing this paper, the feature extraction is manually, while for the classification is used matlab toolbox. The toolbox is classification learner. Next, we load our data and select the gaussian SVM.

They explain that they built three dataset. In this sense, we obtained *three datasets*. The *first dataset* is obtained applied the first formula. This dataset has a matrix of 1080 rows and 76 columns. This data feeds the SVM classifier and training with cross-validation with k-fold = 5 and reports 88.9% accuracy.

The *second dataset* is formed by 1, 4, and 6 formulas. In this sense, we built a matrix of 1080×276, and we get an accuracy of 92%. This value increases concerning the previous one, but it is still below the characteristics' use. Considering that formulas 2 and 3 are not available because we do not have those values in our dataset.

Regarding the *last dataset*, we entered a matrix of 1080 by 500, applying formulas 1, 4, 5, 6, and 7. We also trained the classifier with cross-validation, as indicated in the paper. The results were 94.4%.

However, they report 99.58% as an average of all models. Also, reported variance of $+-$ 6%.

In the Paper [43], the author mentions that machine learning algorithms usually require the extraction of certain features from their respective classification data. The author mentions that one of the features used in his research was 5 normalized distances from each finger's tips to the center of the palm. These distances will later be called tip-to-palm distances (TTP). Besides, he mentions that the distances were normalized by dividing each distance by the palm's maximum TTP distance, thus obtaining values between 0 and 1.

The middle finger's maximum distance is known as the scale factor and is recorded at the beginning of the system. The feature vector will contain five normalized TTP distances corresponding to each corresponding finger. The normalization process divides each finger's TTP distance by the maximum distance of the same finger, instead of the middle finger. The feature vector will thus describe the proportion by which each finger of the palm is extended.

During the calibration process, the user is asked to hold their hand extended in front of the LMC. The length of each extended finger is measured, and the system records the TTP distance of each finger from this display. This measurement vector is the user's hand template, and each gesture performed thereafter will have its TTP distances divided by the template. The user's hand template is specific to a given user and is recorded once per session. As shown in Fig. 12.

Subsequently, the use of these 5-dimensional feature vectors makes the gestures independent. Dimensions make the gestures independent of the orientation and are orientation and can accurately describe the hand posture. The posture of the hand.

Also, the researchs mention that calculating the magnitude of a vector requires a square root operation, which can be computationally expensive, especially if performed during each application update step. To alleviate this problem,

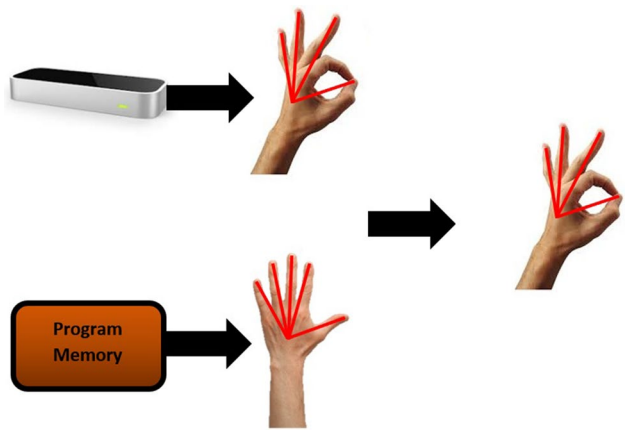

Fig. 12 It represents the values of the template of the fully stretched hand. Also, it presents the distance from the fingers to the center of the hand by executing a gesture. These values are compared with the template, and the gesture is determined

they use the square of the TTP distance's magnitude to approximate the square root. This also means that the user's hand template's magnitudes will have to be squared so that the normalized distances will no longer be scaled linearly. However, by avoiding a square root operation, system performance is improved.

To achieve this, the first repetition of the first OpenHand gesture was used in which its frames are traversed and the distance from the center of the palm to each finger is calculated in each of these frames. Having these distances in the matrix M ($5 \times$ NFrames), the maximum value of each of the columns is searched within the matrix, this being the maximum TTP distance of each finger $t = max(M)$, t returns a 1×5 vector containing the user's maximum TTP distances.

This vector contains the maximum TTP distances for the user and will be used as your base template for the following dataset calculations.

To obtain the feature vectors of each gesture with its label, we run through each of the repetitions of the gestures performed by the user and generate the 5-feature vector that indicates the base study, these five normalized values represent the percentage of elongation of each finger (d) in the current pose.

Once the matrix containing the feature vectors of each gesture has been formed, the frame that symbolizes the gesture is searched for. For this, we look for the frame whose TTP distance of all the fingers on average is the smallest compared to those of the other frames, being this the vector of characteristics of the gesture. In this context, the gesture is found in the frame where the TTP distances are the most different from those of the template. This dataset feeds the classifier KNN. For the reproducibility of the paper, we use the Matlab toolbox. The paper has an accuracy of 82.5%. They point out that for testing, they present a new dataset. In this sense, we also present data from other users for reproducibility, and an accuracy of 87.5% is presented. However, testing the same model for more users (10), the accuracy drops to 72%. This shows that the model does not generalize.

In Ref. [31] the authors used the spatial positions and velocities of the hand. One frame contains for each finger the spatial positions and velocities in x, y, z, i.e. 5 $(3+3)=30$ positions. It has 97 recordings where each recording has 1812 frames representing a sequence of six gestures. In this sense, we create a dataset with eight gestures and 56 users. Each user performs 30 repetitions of each gesture. To form the dataset, we select the gestures close hand, wave in, wave out, pinch, circle, and swipe. Besides, preprocessing is performed because the number of frames is different in each repetition, and a pre-determined number of 70 frames is set. Also, the gestures are manually segmented, and the frames are labeled as inside and outside. From each frame, we obtain the spatial positions and the velocity in x, y, and z,

obtaining a 70 × 32 matrix. Then, the matrices corresponding to the six gestures and all the users are concatenated to simulate the original paper's dataset.

The authors mention the training of an LSTM network for gesture detection. In this context, the LSTM network is set up with 30 features, 125 hidden layers, where the output is a single value. This output is given by linear regression. The number of training epochs is 120, the minibatch size is 27, and the loss function is Adam, and the gradient threshold equal to 1. This predicts a label that is stored in the same dataset by adding a column.

These labels support post-processing because the gesture detection model identifies a small group of frames as outside is the true inside label. The post-processing consists of going through the dataset and obtaining the matrix of the six gestures, taking for each gesture the values of the corresponding rows and columns. First, from each gesture, the rows' indexes with a value of 1 in column 32 are extracted. If the index of position i + 1 minus the index of position i is less than 5, then those segments are labeled as inside.

Second, from each gesture, the rows' indexes containing the value 0 in column 32 are extracted. These values are traversed from j = 1 to the maximum amount minus 1. If the indices of position j + 1 minus the index of position j are less than and equal to 10, those segments are labeled as outside. Third, for each gesture, the indexes of the rows having the value 1 in column 32. For the first index and last index, these values are duplicated 10 times. The 10 duplicates of the first index are added to the segment's left side, and the remaining 10 duplicates are added to the right side of the segment. Added to the right side of the segment. To summarize, if the segments labeled inside are separated by rows labeled outside by a value less than and equal to 5, those rows are labeled as inside. If the segments labeled as inside have several frames less than 10, they are labeled as outside. The first and last frames are duplicated and added to the segment's left and right sides, respectively.

The authors input the dataset to an autoencoder network of 50 neurons in the hidden layer to reduce the number of input features to the LSTM neural network with the same configuration used for detection. With the difference that a softmax gives the output. Besides, the classifier is trained with cross-validation with k = 10; and 150 epochs. Average accuracy of 56.93% is reported.

In Ref. [64], the authors present a new deep neural network architecture called HBU-LSTM for hand gesture recognition using hand motion's temporal data sequences. The idea comes from combining the U-LSTM layer and a Bi-LSTM layer. In the proposed architecture, the Bi-LSTM layer is the first to be used. It is the one that learns the features and delivers them to an LSTM network that acts as an intermediate layer. This layer is connected to the U-LSTM layer. Between these two layers, a dropout layer is implemented to avoid overfitting. Finally, it is connected to a fully connected layer, which is also preceded by a dropout layer. The activation function in the output layer is a softmax, giving the resulting class probability. To carry out this proposal, the authors use a dataset called LeapGestureDB. This dataset consists of 120 users, each user represented in a directory. Each directory contains files in.txt format representing the 11 proposed gestures. Each gesture is repeated 5 times. In other words, there are 55.txt files in each directory. As the dataset was released, we downloaded it to perform the reproducibility of the paper. Each file of the dataset has a different size and weight, but some files have a length much lower than the average weight of the other files. In this sense, for our process, we exclude files larger than 500,000 bytes. This shows that these files have had a longer sampling time. As a pre-processing, we normalize the data between [− 1, 1]. The resulting dataset is obtained through the program extractData.mat in which the.txt files are read from each directory, and the coordinates [X, Y, Z] of the six most dominant points of the hand are extracted: the center of the palm P and the five fingertips F1, F2, F3, F4, F5. The values of interest are obtained by regular expressions (regexp) and normalized with the Matlab normalize function. The labels of each gesture are extracted from the file name with the regexp function. The dataset structure consists of a vector of 1 × K cells where K is the number of users of the database. Each cell contains a matrix of N x M where N is the total number of frames of the five repetitions of the 11

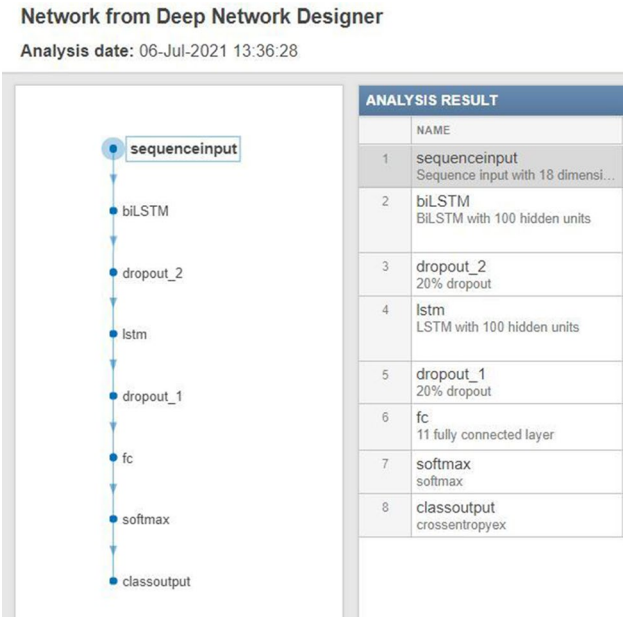

Fig. 13 Representation of the classifier architecture using the Matlab toolbox called the network designer

gestures and M are the six points of the hand in the [X, Y, Z] axes plus the label to which it corresponds giving a total of 19 columns.

The HUB-LSTM neural network is developed using Matlab's deepNetworkDesigner function. As shown in Fig. 13. This network contains a sequential input layer, a BI-LSTM layer, and a dropout layer. A U-LSTM layer and another dropout layer are also added, followed by a FullyConnectedLayer with 11 hidden neurons, a Softmax layer, and finally, a classification output layer. The dropout layer has a probability of 0.2, and all LSTMs have 100 hidden neurons. For the network's hyperparameters, a batch size of 30 with a maximum of 200 epochs, the Adam optimizer, and an initial learning rate of 0.005 is used. The entire configuration is adopted empirically.

The experiment is performed on an Asus Intel Core i7-9750H CPU with 2.60 GHz and 16 GB RAM with Matlab R2018b. The data were empirically divided into [70/30] for training and evaluation, respectively. The posed network has 86.20% accuracy, while the original work obtains 89.98% accuracy.

The results obtained are similar to the original work, with a difference of 3.78%. This difference may be due to how the authors obtained the data based on a previous work where, in addition to the data mentioned, they calculated the arithmetic mean, the standard deviation, and the covariance. The root means square when $Wt = 20$. Another factor that can interfere in the result is that in some cases, the regexp function does not obtain the values of the total number of frames of each of the points due to unknown circumstances, so to correct this error, we take the data corresponding to the lowest number of frames obtained, that is, if from the six points of interest we obtain L number of frames where L is a vector containing the number of rows of each of the points of interest, we determine the lowest value of L (Min) and take the data of all the points from 1 to Min.

5 Discussion

In this context, one of the biggest challenges for researchers starting work with machine learning is to obtain a dataset that represents the problem they are investigating. In this sense, for reproducibility and analysis, papers have been chosen that for their development acquire data using the Leap Motion Controller because this device returns a large amount of information in a very useful time series for gesture recognition in non-invasive systems. Also, in the analysis, it is important to note that the authors propose to carry out preprocessing. Due to the characteristics of the Leap Motor Controller data, one of the techniques used is normalization. They also use techniques to smooth the signal, such as the wavelet transform. This function can work both in the time domain and in the frequency domain and works as a low-pass and high-pass filter simultaneously. Likewise, the authors present the dimensionality reduction technique using principal component analysis or autoencoder neural networks.

The authors also present feature extraction techniques. Based on the works analyzed and reproduced, it is considered that this process is of great importance for obtaining high accuracy in the process of classification and recognition of hand gestures. It can also be mentioned that the features are in relation to the types of gestures that the authors propose for recognition, i.e., for some gestures, certain features may adequately represent the gesture, while for other gestures, they do not. As in the case of Ref. [50], who propose a feature extractor with which they obtain a high accuracy value. But in the case of the dataset that we used for the reconstruction of the papers, in the wave in and wave out gesture, the features such as the distance between the tips of the adjacent fingers, the angle between the first finger and each of the other fingers, the distance between the fingertip and the center of the palm, the angle between the adjacent fingers are not adequate, because by the nature of these gestures these features do not vary, they remain almost constant, and that makes that high accuracy is not achieved. In the same sense, in the case of Ref. [43] that uses a pattern gesture to measure the distances between the fingertips and the center of the palm. This feature may be useful for gestures in which the fingers contract towards the center of the palm, as demonstrated by them, while for gestures in which the fingers remain stretched, this feature is not relevant. In this sense, it is difficult to propose a feature extractor that can generalize all problems.

In this context, many authors propose to perform feature extraction automatically, using the concept of deep neural networks, such as CNN or LSTM. In Refs. [64, 65], the exposed above is evidenced by the same type of gestures presented in the LeapGestureDB dataset. They propose a manual feature extraction method called chronological indexing. They also present automatic feature extraction using the unidirectional and bi-directional LSTM join method.

It is also important an adequate selection of the classification algorithms. Several classification algorithms are used, and each of these with its respective parameters. In this sense, we cite KNN. This is a non-parametric classifier that is necessary to configure the number of neighbors, the distance method. Many authors use DTW as a distance measure because of the number of parameters that can be configured. Another widely used classifier is SVM. This classifier is a binary classifier. With the one vs. one or one vs. all techniques implemented, it becomes a multiclass classifier. It also implements linear or radial basis functions. ANNs are

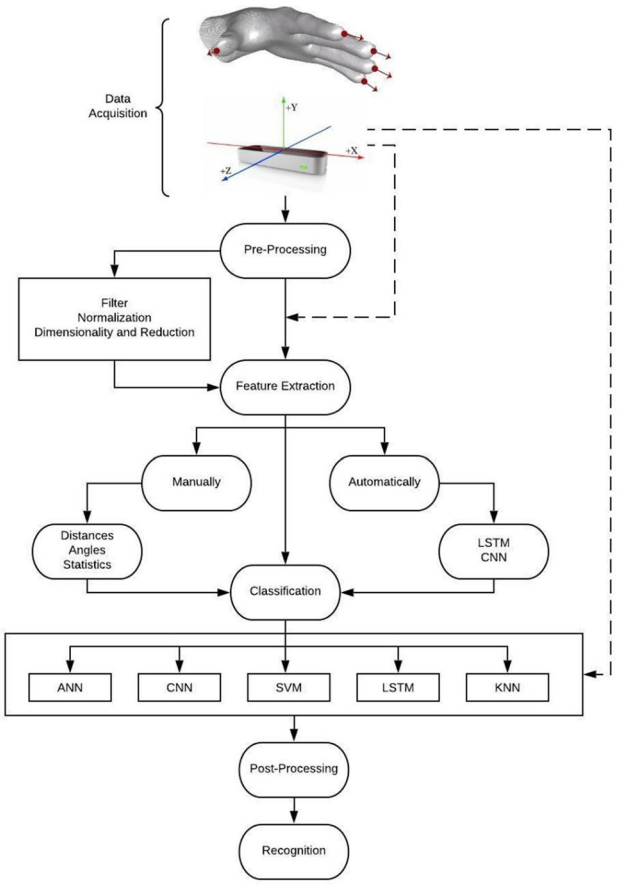

Fig. 14 Represents the generic model consisting of data acquisition, preprocessing, feature extraction, classification, and postprocessing modules for hand gesture recognition based on infrared information and machine learning algorithms

also implemented in the hand gesture classification problem. In this type of algorithm, it is feasible to configure the number of hidden layers, the number of neurons per layer, the activation function, the cost function optimization method, the number of training epochs.

For the classification of dynamic gestures, some authors use algorithms such as LSTM or CNN. It is also necessary to configure parameters such as the number of hidden layers, convolution masks, and activation functions to these algorithms.

It is important to note that the accuracy also depends on the type of parameters with which each classifier is configured.

In this context, a representation of the generic model architecture is shown in Fig. 14, the modules and techniques used for hand gesture recognition using infrared information and machine learning algorithms.

6 Conclusions and gaps

The study presents an exhaustive systematic literature review, introduction to the problem, a summary of systematic reviews regarding the issue. Besides, this work describes the variables involved in the construction of the research questions these are population, intervention, and outcomes. Identified the keywords and synonyms for building the search strings, the scientific databases for searching primary studies, the inclusion and exclusion criteria are detailed, the criteria for quality assessment of the selected articles, this avoiding bias in the research. It also presents a section of experimentation, where four articles are reproduced. These articles are reproduced because they are the ones that use most of the techniques mentioned in the development of the article. Finally, present the discussion of the results.

It is concluded that the scientific literature presents several models that try to solve the problem of hand gesture recognition using machine learning and infrared information, each of these presents results of their research the value of accuracy of the classification and some works presents the processing time. However, these works are not comparable because the models evaluate different types and number of gestures, the construction of the dataset differs in the number of samples obtained, the shape, origin, and type of sensor used. The models also present different techniques used in the modules of data acquisition, pre-processing, feature extraction, classifiers, and post-processingIn this sense, approached the problem from the type of supervised learning and that the adjustment of parameters in the modules of data acquisition and pre-processing is carried out in a heuristic way and by trial and error, the module of feature extraction is performed automatically, of heuristic mode and while the classification module is automatic.

As gaps, we can mention the deficiency in data acquisition protocols, the construction of the generic dataset that allows researchers to test their models, and measure the classification accuracy, recognition, and processing time of the algorithms. Also, there are datasets with a limited number of samples and few repetitions of the gestures acquired with Leap Motion. The lack of automatic feature extraction methods for spatial positions, classifiers from the family of recurrent neural networks for data acquired with Leap Motion. Finally, he concluded that obtaining a classifier that reports high accuracy and processing time remains the biggest challenge in the field of hand gesture recognition using machine learning and infrared information.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s13042-021-01372-y.

Acknowledgements The authors' gratitude to the Escuela Politécnica Nacional and its doctoral program in computer science. For having

the best human resources for the development of its students. Thanks are also due to the Universidad Técnica de Ambato, for providing the facilities for continuous improvement.

Funding The Escuela Politecnica Nacional supported the development of this work through computer science faculty and the informatics doctoral program.

References

1. Chaudhary A, Raheja JL, Das K, Raheja S (2013) Intelligent approaches to interact with machines using hand gesture recognition in natural way: Survey. arxiv:1303.2292
2. Mcintosh J, BI Group (2017) SensIR: detecting hand gestures with a wearable bracelet using infrared transmission and reflection. In: Proceedings of the 30th annual ACM symposium on user interface software and technology, pp 593–597. https://doi.org/10.1145/3126594.3126604
3. Pisharady PK, Saerbeck M (2015) Recent methods and databases in vision-based hand gesture recognition: a review. Comput Vis Image Underst 141:152–165. https://doi.org/10.1016/j.cviu.2015.08.004
4. Ren Z, Meng J, Yuan J (2011) Depth camera based hand gesture recognition and its applications in human-computer-interaction. In: 2011 8th international conference on information, communications & signal processing, pp 1–5. https://doi.org/10.1109/ICICS.2011.6173545
5. Kumari N, Garg R, Aulakh IK (2014) A spiking neuron improved PCA model for hand gesture recognition. In: ACM international conference proceeding series, vol 11. https://doi.org/10.1145/2677855.2677876
6. Dominio F, Donadeo M, Marin G, Zanuttigh P, Cortelazzo GM (2013) hand gesture recognition with depth data. In: Proceedings of the 4th ACM/IEEE international workshop on analysis and retrieval of tracked events and motion in imagery stream, pp 9–16. https://doi.org/10.1145/2510650.2510651
7. Ali HH, Moftah HM, Youssif AAA (2017) Depth-based human activity recognition: a comparative perspective study on feature extraction. Futur Comput Inform J. https://doi.org/10.1016/j.fcij.2017.11.002
8. Al-Khalifa HS (2017) CHEMOTION: a gesture based chemistry virtual laboratory with leap motion. Comput Appl Eng Educ 25(6):961–976. https://doi.org/10.1002/cae.21848
9. Benalcázar ME et al (2017) Real-time hand gesture recognition using the Myo armband and muscle activity detection. In: 2017 IEEE second Ecuador technical chapters meeting (ETCM), pp 1–6. https://doi.org/10.1109/ETCM.2017.8247458
10. Suarez J, Murphy RR (2012) Hand gesture recognition with depth images: a review. In: 2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication, IEEE, pp 411–417. https://doi.org/10.1109/ROMAN.2012.6343787
11. Li G et al (2019) Hand gesture recognition based on convolution neural network. Clust Comput 22:2719–2729. https://doi.org/10.1007/s10586-017-1435-x
12. Pinto RF, Borges CDB, Almeida AMA, Paula IC (2019) Static hand gesture recognition based on convolutional neural networks. J Electr Comput Eng. https://doi.org/10.1155/2019/4167890
13. Sharma A, Mittal A, Singh S, Awatramani V (2020) Hand gesture recognition using image processing and feature extraction techniques. Procedia Comput Sci 173(2019):181–190. https://doi.org/10.1016/j.procs.2020.06.022
14. Visconti P, Gaetani F, Zappatore GA, Primiceri P (2018) Technical features and functionalities of Myo armband: an overview on related literature and advanced applications of myoelectric armbands mainly focused on arm prostheses. Int J Smart Sens Intell Syst 11(1):1–25. https://doi.org/10.21307/ijssis-2018-005
15. Kitchenham B (2004) Procedures for performing systematic reviews. Keele, UK, Keele University. 33(2004), 1–26. ISSN:1353-7776
16. Madeo RCB, Lima CAM, Peres SM (2016) Studies in automated hand gesture analysis: an overview of functional types and gesture phases. Lang Resour Eval. https://doi.org/10.1007/s10579-016-9373-4
17. Groenewald C, Anslow C, Islam J, Rooney C, Passmore P, Wong W (2016) Understanding 3D mid-air hand gestures with interactive surfaces and displays: a systematic literature review, pp 1–13. https://doi.org/10.14236/ewic/HCI2016.43
18. Al-shamayleh AS, Ahmad R, Abushariah MAM (2018) A systematic literature review on vision based gesture recognition techniques. Multimedia Tools and Appl 77(21):28121–28184. https://doi.org/10.1007/s11042-018-5971-z
19. Sathyanarayana S (2015) Vision-based patient monitoring: a comprehensive review of algorithms and technologies. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-015-0328-1
20. Chouhan SS, Kaul A (2018) Soft computing approaches for image segmentation: a survey. Multimedia Tools and Appl 77(21):28483–28537. https://doi.org/10.1007/s11042-018-6005-6
21. Dqj X et al (2014) A novel feature extracting method for dynamic gesture recognition based on support vector machine. In: 2014 IEEE international conference on information and automation (ICIA), pp 437–441
22. Jais HM, Mahayuddin ZR, Arshad H (2015) A review on gesture recognition using Kinect. In: 5th international conference on electrical engineering and informatics 2015, pp 594–599. https://doi.org/10.1109/ICEEI.2015.7352569
23. Plouffe G, Cretu A-M (2016) Static and dynamic hand gesture recognition in depth data using dynamic time warping. IEEE Trans Instrum Meas 65(2):305–316. https://doi.org/10.1109/TIM.2015.2498560
24. Czuszynski K, Ruminski J, Wtorek J (2017) Pose classification in the gesture recognition using the linear optical sensor. In: Proceedings of 2017 10th international conference on human system interactions, pp 18–24. https://doi.org/10.1109/HSI.2017.8004989
25. Park S, Ryu M, Chang JY, Park J (2014) A hand posture recognition system utilizing frequency difference of infrared light. In: Proceedings of the 20th ACM symposium on virtual reality software and technology, pp 65–68. https://doi.org/10.1145/2671015.2671114
26. Jangyodsuk P, Conly C, Athitsos V (2014) Sign language recognition using dynamic time warping and hand shape distance based on histogram of oriented gradient features. In: Proceedings of the 7th international conference on PErvasive technologies related to assistive environments-PETRA'14, pp 1–6. https://doi.org/10.1145/2674396.2674421
27. Doan HG, Vu H, Tran TH (2015) Recognition of hand gestures from cyclic hand movements using spatial-temporal features. In: ACM international conference proceeding series, vol. 03, pp 260–267. https://doi.org/10.1145/2833258.2833301
28. Lu W, Tong Z, Chu J (2016) Dynamic hand gesture recognition with leap motion controller. IEEE Signal Process Lett 23(9):1188–1192. https://doi.org/10.1109/LSP.2016.2590470

29. Wang J, Liu T, Wang X (2020) Human hand gesture recognition with convolutional neural networks for K-12 double-teachers instruction mode classroom. Infrared Phys Technol 111:103464. https://doi.org/10.1016/j.infrared.2020.103464
30. Brock H, Sabanovic S, Nakamura K, Gomez R (2020) Robust real-time hand gestural recognition for non-verbal communication with tabletop robot Haru. In: 2020 29th IEEE international conference on robot and human interactive communication (RO-MAN), pp 891–898. https://doi.org/10.1109/RO-MAN47096.2020.9223566
31. Naguri CR, Bunescu RC (2018) Recognition of dynamic hand gestures from 3D motion data using LSTM and CNN architectures. In: Proceedings of 2017 16th IEEE international conference on machine learning and applications (ICMLA), vol. 2018, pp 1130–1133. https://doi.org/10.1109/ICMLA.2017.00013
32. Benmoussa M, Mahmoudi A (2018) Machine learning for hand gesture recognition using bag-of-words. In: 2018 international conference on intelligent systems and computer vision (ISCV), vol. 2018, pp 1–7. https://doi.org/10.1109/ISACV.2018.8354082
33. Vamsikrishna KM, Dogra DP, Desarkar MS (2015) Computer vision assisted palm rehabilitation with supervised learning. IEEE Trans Biomed Eng 63(5):991–1001. https://doi.org/10.1109/TBME.2015.2480881
34. Almasre MA, Al-nuaim H (2016) Recognizing arabic sign language gestures using depth sensors and a KSVM classifier. In: 2016 8th computer science and electronic engineering (CEEC). https://doi.org/10.1109/CEEC.2016.7835904
35. Avola D, Bernardi M, Member S, Massaroni C, Member S (2018) Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures. IEEE Trans Multimed. https://doi.org/10.1109/TMM.2018.2856094
36. Liu F, Du B, Wang Q, Wang Y, Zeng W (2017) Hand gesture recognition using Kinect via deterministic learning, In: 2017 29th Chinese Control and Decision Conference (CCDC), IEEE, pp 2127–2132. https://doi.org/10.1109/CCDC.2017.7978867
37. Ben Khalifa A (2016) A comprehensive leap motion database for hand gesture recognition, In: 2016 7th international conference on sciences of electronics, technologies of information and telecommunications (SETIT), IEEE, pp 514–519. https://doi.org/10.1109/SETIT.2016.7939924
38. Rossol N, Cheng I, Member S, Basu A, Member S (2016) A multisensor technique for gesture recognition through intelligent skeletal pose analysis. IEEE Trans Hum-Mach Syst 46(3):350–359. https://doi.org/10.1109/THMS.2015.2467212
39. Plouffe G, Cretu A (2016) Static and dynamic hand gesture recognition in depth data using dynamic time warping. IEEE Trans Instrum Meas 65(2):305–316. https://doi.org/10.1109/TIM.2015.2498560
40. Bai X, Li C (2018) Dynamic hand gesture recognition based on depth information. In: 2018 international conference on control, automation and information sciences (ICCAIS), pp 216–221
41. Lai K, Yanushkevich SN (2018) CNN + RNN depth and skeleton based dynamic hand gesture recognition. In: 2018 24th international conference on pattern recognition (ICPR), pp 3451–3456
42. Liu X, Li C, Tian L (2017) Hand gesture recognition based on wavelet invariant moments. In: 2017 IEEE international symposium on multimedia, pp 459–464. https://doi.org/10.1109/ISM.2017.91
43. Clark A, Moodley D (2016) A system for a hand gesture-manipulated virtual reality environment. In: Proceedings of annual conference of the South African institute of computer scientists and information technologists- SAICSIT'16, pp 1–10. https://doi.org/10.1145/2987491.2987511
44. Jiang F, Zhang S, Wu S, Gao Y, Zhao D (2015) Multi-layered gesture recognition with Kinect. J Mach Learn Res 16(1):227–254
45. Hsiao D, Sun M, Ballweber C, Cooper S (2016) Proactive sensing for improving hand pose estimation. In: Proceedings of the 2016 CHI conference on human factors in computing systems, pp 2348–2352. https://doi.org/10.1145/2858036.2858587
46. Ye Y, Nurmi P (2015) Gestimator—shape and stroke similarity based gesture recognition categories and subject descriptors. In: Proceedings of the 2015 ACM on international conference on multimodal interaction, pp 219–226. https://doi.org/10.1145/2818346.2820734
47. Tang AO, Lu KE, Wang Y, Huang JIE, Li H (2015) A real-time hand posture recognition system using deep neural networks. ACM Trans Intell Sys Technol (TIST) 6(2):1–23. https://doi.org/10.1145/2735952
48. Wang C, Liu Z, Zhu M, Zhao J, Chan S (2017) A hand gesture recognition system based on canonical superpixel-graph. Signal Process Image Commun. https://doi.org/10.1016/j.image.2017.06.015
49. Núñez JC, Cabido R, Pantrigo JJ, Montemayor AS, Vélez JF (2018) Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition. Pattern Recognit 76:80–94. https://doi.org/10.1016/j.patcog.2017.10.033
50. Cho Y, Lee A, Park J, Ko B, Kim N (2018) Enhancement of gesture recognition for contactless interface using a personalized classifier in the operating room. Comput Methods Programs Biomed 161:39–44. https://doi.org/10.1016/j.cmpb.2018.04.003
51. Kumar P, Gauba H, Roy PP, Dogra DP (2017) A multimodal framework for sensor based sign language recognition. Neurocomputing 259:21–38. https://doi.org/10.1016/j.neucom.2016.08.132
52. Hong C, Zeng Z, Xie R, Zhuang W, Wang X (2018) Domain adaptation with low-rank alignment for weakly supervised hand pose recovery. Signal Process 142:223–230. https://doi.org/10.1016/j.sigpro.2017.07.032
53. Dynamic AI, Warping T (2016) An image-to-class dynamic time warping approach for both 3D static and trajectory hand gesture recognition. Pattern Recognit. https://doi.org/10.1016/j.patcog.2016.01.011
54. Liang W, Guixi L (2015) Dynamic and combined gestures recognition based on multi-feature fusion in a complex environment. J China Univ Posts Telecommun 22(2):81–88. https://doi.org/10.1016/S1005-8885(15)60643-4
55. Inoue K, Shiraishi T, Yoshioka M, Yanagimoto H (2015) Depth sensor based automatic hand region extraction by using time-series curve and its application to Japanese finger-spelled sign language recognition. Procedia Comput Sci 60(1):371–380. https://doi.org/10.1016/j.procs.2015.08.145
56. Yang J, Horie R (2015) An improved computer interface comprising a recurrent neural network and a natural user interface. Procedia Comput Sci 60:1386–1395. https://doi.org/10.1016/j.procs.2015.08.213
57. Tao W, Leu MC, Yin Z (2018) American sign language alphabet recognition using convolutional neural networks with multiview augmentation and inference fusion. Eng Appl Artif Intell 76:202–213. https://doi.org/10.1016/j.engappai.2018.09.006
58. Leite DQ, Duarte JC, Neves LP, De Oliveira JC, Giraldi GA (2017) Hand gesture recognition from depth and infrared Kinect data for CAVE applications interaction. Multimed Tools Appl. https://doi.org/10.1007/s11042-016-3959-0
59. Marin G, Dominio F, Zanuttigh P (2016) Hand gesture recognition with jointly calibrated leap motion and depth sensor. Multimed Tools Appl. https://doi.org/10.1007/s11042-015-2451-6
60. Jiang X, Gang Z, Carlo X (2018) Virtual grasps recognition using fusion of Leap Motion and force myography. Virtual Real 22(4):297–308. https://doi.org/10.1007/s10055-018-0339-2

61. Quesada L, López G, Guerrero L (2017) Automatic recognition of the American sign language fingerspelling alphabet to assist people living with speech or hearing impairments. J Ambient Intell Humaniz Comput 8(4):625–635. https://doi.org/10.1007/s12652-017-0475-7
62. Ma C, Wang A, Chen G, Xu C (2018) Hand joints-based gesture recognition for noisy dataset using nested interval unscented Kalman filter with LSTM network. Vis Comput 34(6):1053–1063. https://doi.org/10.1007/s00371-018-1556-0
63. Lee GC, YehF, Hsiao Y (2016) Kinect-based Taiwanese sign-language recognition system. Multimedia Tools Appl 75(1):261–279. https://doi.org/10.1007/s11042-014-2290-x64
64. Ameur S, Ben Khalifa A, Bouhlel MS (2020) A novel hybrid bidirectional unidirectional LSTM network for dynamic hand gesture recognition with leap motion. Entertain Comput 35:100373. https://doi.org/10.1016/j.entcom.2020.100373
65. Ameur S, Ben Khalifa A, Bouhlel MS (2020) Chronological pattern indexing: an efficient feature extraction method for hand gesture recognition with leap motion. J Vis Commun Image Represent 70:102842. https://doi.org/10.1016/j.jvcir.2020.102842
66. Samanta D, Panchal G (2016) Advances in soft computing. In: Soft Computing Applications in Sensor Networks, CRC Press. P 21
67. Hisham B, Hamouda A (2019) Supervised learning classifiers for Arabic gestures recognition using Kinect V2. SN Appl Sci. https://doi.org/10.1007/s42452-019-0771-2
68. Hu B, Wang J (2020) Deep learning based hand gesture recognition and UAV flight controls. Int J Autom Comput 17(1):17–29. https://doi.org/10.1007/s11633-019-1194-7
69. Ricci E, Rota S, Snoek C, Lanz O, Goos G (2019) Processing—ICIAP 2019, image analysis and processing–ICIAP 2019. In: 20th International Conference, Trento, Italy, September 9–13, 2019, Proceedings, Part II, vol. 11752. Springer Nature. https://doi.org/10.1007/978-3-030-30645-8
70. Benalcázar ME (2019) Machine learning for computer vision: a review of theory and algorithms. Revista Ibérica de Sistemas e Tecnologias de Informação (E19):608–618