Skin Color Segmentation and Leveshtein Distance Recognition of BSL Signs in Video

Beatriz Tomazela Teodoro, Joao Bernardes and Luciano Antonio Digiampietri
School of Arts, Sciences and Humanities, University of São Paulo,
São Paulo, Brazil
{beatriz.teodoro, jlbernardes, digiampietri}@usp.br

***Abstract*—Sign language automatic recognition is an important research area with open challenges that aims to mitigate the obstacles in the daily lives of people who are deaf or hard of hearing and increase their integration in the predominantly hearing society in which we live. This paper implements, evaluates and discusses strategies for automatic recognition of Brazilian Sign Language (BSL) signs, which ultimately aims to simplify the communication between deaf signing in BSL and listeners who do not know this sign language, accomplished through the processing of digital videos of people communicating in BSL without the use of colored gloves or data gloves and sensors or the requirement of high quality recordings in laboratories with controlled backgrounds or lighting. An approach divided in several stages was developed and all stages of the proposed system can be considered contributions for future works in sign language recognition or those involving image processing, human skin segmentation, object tracking etc. For the skin color based segmentation stage, in particular, several techniques were implemented and compared and the strategy used for sign recognition, exploring the Leveshtein distance and a voting scheme with a binary classifier, is unusual in this area and showed good results. From the original 600 samples of 30 words, chosen for frequency of use and superposition of sign elements to make recognition more complex, the system was able to correctly segment 422 (70%) signs, for which it reached 100% accuracy in recognition using our strategy. This sign database with 600 samples in video of the chosen 30 word vocabulary is another of this work's contributions and is available upon request to the authors.**

***Index Terms*—sign language recognition; image processing; human skin segmentation; Brazilian Sign Language; LIBRAS**

I. Introduction

The deaf community faces a major obstacle in their daily routine: the difficulty in communicating with the predominantly hearing society. In recent years, there has been an increasing commitment to facilitate communication between deaf or hard of hearing people with people who do not know a sign language, but there are still few accessible environments and systems for that. The lack of sign language knowledge in the general population makes the communication with deaf people extremely difficult. Moreover, recognition and translation of sign languages by computers are quite complex areas, in which most studies are recent and present open challenges [1], [2], [3].

In sign languages, either Brazilian or others [4], [5], [6], there are some static signs to indicate words and letters, but many of the signs are dynamic, involving not only static hand configurations, but also their movements and changes in configuration. Thus, recognizing these signs automatically requires processing sequences of hand poses instead of only individual configurations.

There are two main approaches to signal acquisition in the sign language recognition area: the vision-based approach and the sensor-based approach. In the first, data is obtained through one or more video cameras; it is more convenient for the user, but requires sophisticated image processing. In the second approach, data is obtained through electromechanical devices, such as data gloves, or other sensors, bringing limitations and discomfort to the user [3], but facilitating recognition, avoiding problems faced in the first approach, such as segmentation of the hands in images [2]. In order to decrease image processing complexity in the vision-based approach, several studies make use of controlled environments and/or colored gloves when recording the videos that will be analyzed, which considerably facilitates image processing, segmentation and the subsequent sign recognition.

In our work, however, we opt for recognizing dynamic Brazilian Sign Language (BSL) signs using a vision-based approach with a single 2D camera and without taking advantage of colored gloves or high quality video recordings made in labs with controlled environments. BSL, or LIBRAS, is the official language used by the deaf community in Brazil. Integrating these image processing and sign recognition strategies within more complex grammar and language translation systems in the future, we ultimately aim to simplify the communication between deaf people signing in BSL and listeners who do not know this sign language. Given the complexity of vision-based recognition without colored gloves or controlled environments, this approach was implemented in several stages. Its main focus was in two areas: investigating several alternatives for hand segmentation based on skin color, either using machine learning or explicit rules; and identifying and translating dynamic BSL signs in the segmented image sequences, using Leveshtein distance.

This work also brings an additional contribution: an annotated database with 600 video samples of ten different BSL users, each performing two samples of thirty specific BSL signs. These signs were selected based on two criteria: high frequency of use in the language and existing superposition of hand configurations and movements between the different signs, to make the recognition task more complex. This video database is available for future use in research by contacting

any of the authors (it has not been made public a priori only due to concerns of our Ethics in Research Committee). The vocabulary size for this database is compatible with several of the similar works discussed in the next section. It is worth mentioning that, due to the challenges found in sign language recognition and language translation in general, our goal is not as ambitious as implementing a complete and permanent solution to the automatic translation of BSL, but it is rather an extensible tool to assist in this process. Furthermore, none of the assumptions, algorithms or developed systems and subsystems described here are specific for BSL, so they could probably be used for any sign language, but all results presented in this paper considered only BSL and the sign database that was created includes samples only of this language.

This paper is organized as follows. After this introduction, Section II discusses the state-of-the-art in sign language recognition. Section III presents the image database built for this work. Section IV discusses the strategies, algorithms and system investigated in this study and their implementation. Experimental results are presented in Section V. Finally, Section VI summarizes the main conclusions and discusses future work.

II. Related work

Sign language recognition has been extensively studied and a vast literature has been produced. In order to assess the state-of-the-art in sign language recognition, we performed a systematic review focusing on papers which use the visual approach, i.e., when the data is obtained by video cameras, without the use of other sensors (such as data gloves), identifying the major challenges of working with this approach.

The search was performed in three digital libraries which index the papers published in the main conferences and journals of the area: the ACM Digital Library[1], the Brazilian Digital Library of Computing (BDBComp)[2] and IEEExplore[3]. In addition to these three libraries, papers were also searched in all published and available proceedings of the Symposium on Virtual and Augmented Reality (SVR), the Brazilian Symposium on Artificial Intelligence (SBIA) and the SIBGRAPI Conference on Graphics, Patterns and Images. From this search, 166 papers were obtained. Out of these, 90 were selected for more detailed analysis based on the review protocol's inclusion and exclusion criteria. Due to space limitations, this paper presents only a brief summary of this review, highlighting the main features of the selected works about sign language recognition.

In the majority (80%) of selected papers, colored gloves are not used when capturing the videos and, therefore, the proposed solutions required sophisticated image processing approaches. In order to facilitate image processing and, consequently, sign recognition, some work took advantage of restrictions in their captured videos. Some impose the restriction that the person performing the signs must use a long, dark colored shirt [7], [8], [9]. Hienz *et al.* [10] present a different approach. Besides using multi-colored gloves for recording the videos used in the tests, colored markers were also placed on shoulders and elbows to further reduce recognition complexity. Dimov *et al.* [11] use an even more intrusive restriction. Sign language users perform the signs in a controlled environment, covered with a dark cloth so only their head and hands showed.

Many papers do not discuss the image processing techniques used, focusing instead on the recognition strategy, but out of those who discussed this, segmentation of the regions of interest (ROI, the hands) was often the main focus and most complex problem. Thresholding based on explicit rules for skin color was the most frequently used technique to segment images [12], [13], [8], followed by techniques using Gaussian Mixture Models (GMM) [14], [15], [16]. Some papers took advantage of machine learning algorithms in order to perform image segmentation. Han *et al.* [17] used Support Vector Machines (SVM) to segment skin regions, classifying each pixel as skin or not-skin, with an average classification rate of 76.77% using 240 frames from the ECHO database[4]. Van Hieu and Nitsuwat [7] used a Two-Layer Neural Network (TLNN) to approximate a skin model, using the Cb and Cr chromaticity components from over 414 thousand pixels samples, in order to segment the images, achieving an accuracy rate of 94% for those pixels. Disparity map and the K-Means Clustering method are used by El-Jaber *et al.* [18] to segment the ROI, while Gonçalves *et al.* [19] performed the segmentation using a Multilayer Perceptron (MLP) with the backpropagation algorithm, optimized using the Levenberg-Marquardt method, but the accuracy rate of the latter two works were not presented by the authors.

Soontranon *et al.* [20] present a face detection and hand tracking method for a sign language recognition. They use four videos in Thai Sign Language recorded in a studio without the use of gloves, with a resolution of 240x320 pixels. The segmentation of skin regions (face and hands) is made using the eclipse function, formulated through of the Cb and Cr chromaticity components distribution of the image pixels, reaching an average recognition rate of 89.8%.

It was also noted that machine learning algorithms are used for recognition of the gestures in most of the analyzed works, especially Hidden Markov Model (HMM) and Artificial Neural Network (ANN). Starner and Pentland [21], for instance, present an accuracy rate of 99.2% to recognize 40 words in American Sign Language (ASL), using HMM and hand configuration, point of articulation and movement as parameters, but the videos were recorded with a resolution of 320x243 pixels in a studio using colored gloves, which greatly facilitated hand segmentation and recognition of their configuration. ANNs are used to recognize gestures in Malaysian Sign Language (MSL) [9], reaching a success rate of 92.07%, using 32 different gestures recorded in a studio with a resolution of 320x240 pixels, without the use of gloves, but with restrictions on the clothing of the person performing gestures.

[1] dl.acm.org

[2] www.lbd.dcc.ufmg.br/bdbcomp

[3] ieeexplore.ieee.org

[4] www.let.ru.nl/sign-lang/echo

Another point that drew our attention in the analyzed literature is that there are few sign language image databases. Most studies did not specify the image database used (only 18% did), or used private videos, recorded specifically for the research. Of the few specified image databases, none is Brazilian and the majority is of facial expressions.

In this review we also observed that the main results achieved deal only with static signs recognition or recognition of signs recorded with restrictions, such as the use of colored gloves and/or controlled environments, in order to facilitate the recognition process. The segmentation of the region were the gestures are performed is still a complex task, especially without the use of any mechanism to facilitate the recognition, as well as recognition of the gestures in sign language recorded in this way.

III. Image Database

Sign languages are not universal, thus, each country has its own sign language, influenced by national culture. In this work we built a BSL database to test our segmentation and recognition strategies consisting of 600 colored image sequences (videos) recorded by ten different persons (four men and six women) signing in BSL, without the use of colored gloves and/or data gloves and sensors. Each person performed a set of thirty basic words in two different videos, recorded without controlled lighting, and using different backgrounds and clothing, in order to provide greater variability. The thirty signs were chosen by an expert in BSL based on two criteria: high frequency of use in the language and existing superposition of hand configurations and movements between the different signs, to make the recognition task more complex. The videos were recorded using a Sony Cyber-Shot DSC-W530 14.1 MP digital camera, operating in automatic mode and using a tripod. Each video has 29 frames per second with a resolution of 640x480 pixels. This video database is available for future use in research by contacting any of the authors (it has not been made public a priori only due to concerns of our Ethics in Research Committee). Fig. 1 shows some examples of frames that compose the image database built (one may notice the uncontrolled lighting and complex backgrounds).

Fig. 1. Examples of frames from image database.

IV. Implementation

As mentioned in the introduction, given the complexity of the task undertaken in this study, the implementation was divided in several stages, with a focus in segmentation of the ROI, based on skin color, and in sign recognition. In this section, we discuss these main stages and their implementation. Fig. 2 illustrates the main stages implemented in this sign recognition system in order to provide a better overview of the system.

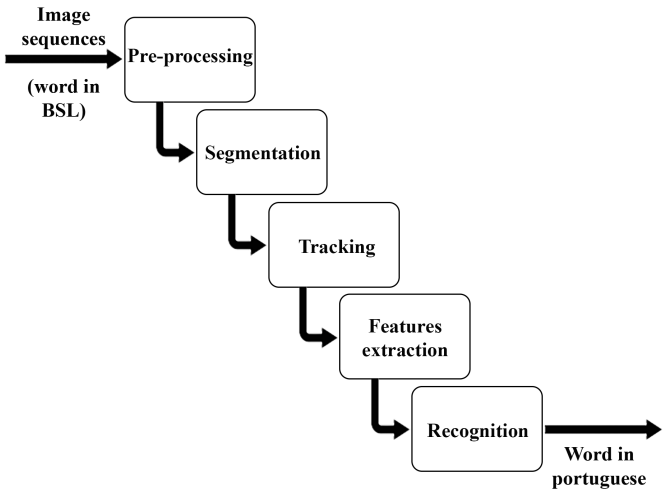

Fig. 2. The main stages of the proposed system.

A. Pre-processing

In this first stage, the main goal is to pre-process the image sequences, segmenting regions with moving objects (in this case, the arms and hands), in order to facilitate the process of identifying and tracking the hands performed later. Towards this end, we used some image processing techniques which are presented in the next subsections.

1) Histogram equalization: Analyzing the image sequences from the image database built to test the proposed system, we noticed the need for some minor adjustments, in order to repair some differences that occurred during image capture, mainly because of environmental lighting differences. Therefore, in order to enhance the contrast of image sequences, first we used the histogram equalization technique. There are several methods that perform histogram equalization. In this work, we used the Contrast Limited Adaptive Histogram Equalization (CLAHE) method.

In this method, instead of calculating a global histogram, a local histogram is calculated considering the neighborhood of each pixel. Moreover, this technique imposes a constraint on the resulting contrast providing a mechanism to handle a possible over-saturation of the resulting image. A maximum number of pixels which are allowed to occupy a bin in the resulting histogram is determined. In cases of over-saturation, the excessive amount of pixels is redistributed over the rest of the histogram [22].

To implement the CLAHE technique we used a function of the "ij.jar" application package from the ImageJ[5], a public domain Java image processing program [23]. The function has three parameters: *block size* (the size of the local region around a pixel for which the histogram is equalized), *histogram bins* (the number of histogram bins used for histogram equalization) and *max slope* (the limit of the contrast stretch in the intensity transfer function). In this work, the parameters were set to the values 63, 256 and 3, respectively.

2) Background subtraction: Background subtraction is one of the main pre-processing steps, widely employed in vision-based applications to separate moving objects (or parts of them) from the rest of the image. Thus, elements of the scene that should not be analyzed can be removed, making the processing faster and often more accurate.

In this work, background subtraction was applied to separate the region of the hands from the static elements of the video, resulting in a second sequence of images containing only the region of the hands. For this, we used a function from OpenCV[6] (*Open Source Computer Vision*) library to implement the background subtraction, the *BackgroundSubtractorMOG2*, where an adaptive model of Mixture of Gaussian (MOG) is internally built for background subtraction with detection of shadows, based on the method of Zivkovic [24] and Zivkovic and van der Heijden [25]. In this method, each pixel is modeled as a MOG, and, in each iteration, the probability that the pixel belongs to the background is calculated. The function has the following parameters: *history* (length of the history), *varThreshold* (threshold on the squared Mahalanobis distance between the pixel and the model to decide whether a pixel is well described by the background model) and *bShadowDetection* (define if the algorithm will detect shadows, with the values "true" or "false"). The parameters were set for this work as 0, 32 and false, respectively. One important feature of this version of the algorithm is the automatic selection of the appropriate number of Gaussian distribution for each pixel, unlike the function *BackgroundSubtractorMOG*, where a fixed number of Gaussian distributions is used throughout the algorithm. Thus, *BackgroundSubtractorMOG2* provides a better adaptability to varying scenes, including to illumination changes.

3) Closing operation: Analyzing the results obtained by the background subtraction technique previously defined, it is possible to observe some noise and defects in the produced images, mainly because of non-static background environments where the videos were recorded. To correct these problems, the closing technique was used.

Closing is a morphological operation used to repair images, defined as a dilation followed by an erosion, using the same structuring element for both operations. This technique tends to smooth sections of contour; it generally fuses narrow breaks and the long thin gulf, eliminates small holes and fills gaps in contour. It fills empty pixels and removes noisy pixels from inside the object, but keeps the shape and size of the object unchanged [26].

In this work, the dilation and erosion filters were performed using the functions *cvDilate()* and *cvErode()* of the OpenCV library, adjusting the following parameters: *src* (source image), *dst* (destination image), *element* (structuring element used for dilation; if it is NULL, a 3x3 rectangular structuring element is used) and *iterations* (number of times dilation or erosion is applied). The dilation filter was applied three times and the erosion fifteen times, with the *element* parameter equal to NULL.

4) Median filter: In order to smooth noise, but preserving the edges and fine details of the produced images after applying the closing operator, the 3x3 median filter was applied, using the function *cvSmooth()* of the OpenCV library.

Fig. 3 shows an example of result obtained by the application of the all pre-processing steps defined in an original image of the image database built.

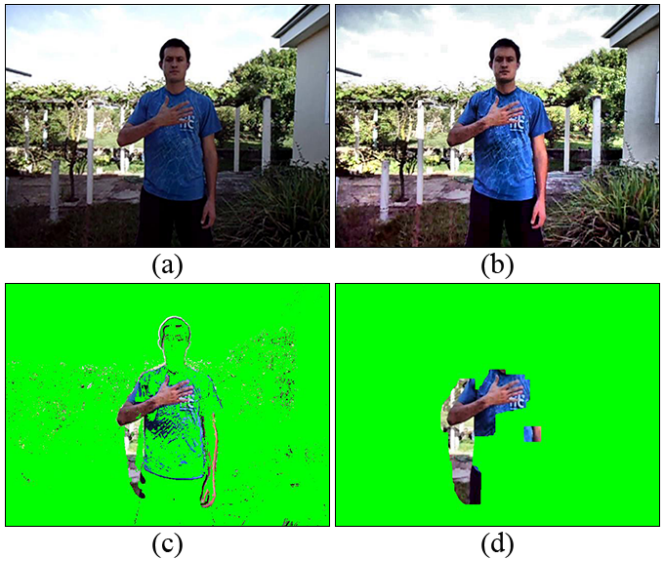

Fig. 3. Results obtained by the pre-processing step: (a) original frame; (b) frame with equalized histogram; (c) frame after applying the background subtraction technique; (d) resulting frame, after applying the closing operation and the median filter.

B. Segmentation

After finishing the pre-processing steps, an algorithm is applied to better segment the ROI in the processed images. This is essential later for the analysis and identification of image features. The goal of this segmentation stage is to identify human skin colored regions, in order to facilitate the detection of hands in the image sequences, preparing them for the extraction of features for sign recognition. To achieve this, the main techniques found in the systematic review that dealt with skin segmentation were implemented. These techniques were tested and the one which obtained the best result was used in this work.

[5]http://imagej.nih.gov/ij/
[6]http://opencv.org

First we tested a collection of machine learning algorithms present in the Weka[7] (Waikato Environment for Knowledge Analysis) software package [27]. The goal of each algorithm was, given the colors of a pixel, return the class of this pixel, classifying it as "skin" or "non-skin" (in this work, "non-skin" will be referred to as "background").

A dataset with 4.000 pixels was built for testing Weka's classifiers. Each pixel instance is composed of four attributes, the *RGB* color components (red, green, blue) and luminance (*Y*), in order to deal with illumination variations on human skin; and, besides these attributes, the label class ("skin" or "background"). To build the data set, 4.000 pixels were randomly extracted (2.000 classified as "skin" and 2.000 as "background") of 200 frames taken from the recorded videos in our sign database. These frames were manually segmented in order to allow the evaluation by the classifiers. The classification algorithms were tested using 10-fold cross-validation and were evaluated using the average values of accuracy, sensitivity and specificity. Accuracy measures the algorithm's ability to correctly determine what is true (in this case, "skin") between all pixels in the image, i.e., how accurate the algorithm is. Sensitivity or recall is the amount of true positives, i.e. how many of the pixels classified as "skin" were actually "skin"; while specificity measures the algorithm ability to properly exclude those pixels that are "background". It would be expected to get high values for sensitivity and specificity for the classifier to correctly identify the pixels that are "skin" and those that are not ("background"), and an accuracy closer to 100%. Table I presents the ten best results achieved by the classifiers in the skin segmentation problem.

TABLE I
CLASSIFIERS TOP TEN RESULTS IN THE SKIN SEGMENTATION PROBLEM

Classifier	Accuracy	sensitivity	specificity
RotationForest	98.25%	99.30%	97.20%
Logistic	98.23%	99.20%	97.25%
MultiClassClassifier	98.23%	99.20%	97.25%
MultilayerPerceptron	98.20%	99.25%	97.15%
ThresholdSelector	98.20%	99.25%	97.15%
SimpleLogistic	98.18%	99.55%	96.80%
LMT	98.17%	99.55%	96.80%
Bagging	98.15%	99.30%	97.00%
RandomForest	98.12%	98.90%	97.35%
NNge	98.08%	98.50%	97.65%

As shown in table I, the *RotationForest (RF)* classifier achieved the best results , with an average rate of accuracy, sensitivity and specificity of 98.25%, 99.30% and 97.20%, respectively. It was, then, the classifier chosen to be used in the recognition system, up to this point.

Because Weka has an easy to use API, the selected algorithm (*RotationForest*) was not reimplemented, instead, we used the implementation available from Weka. We developed a tool that converts the resulting forest from *RotationForest* execution into an expert system that receives the color (*R,G,B*) and the luminance (*Y*) of a given pixel and returns the corresponding class of this pixel ("skin" or "background"). Therefore, the segmentation method calls this expert system to classify each pixel of the image. We will also call this method *RotationForest* in this work in order to simplify its identification. Luminance was calculated using the equation: $Y = (0,299 \times R) + (0,587 \times G) + (0,114 \times B)$.

Besides using the *RotationForest* algorithm to perform skin pixel classification, four other simple human skin segmentation algorithms found in the literature which use a threshold technique based on explicit rules were also implemented [28], [29], [30], [31]. These algorithms receive the color (*R,G,B*) of a given pixel and classifies it as "skin" or "background" based on a set of rules. The test results of the five implemented segmentation methods will be presented in Section V.

The first algorithm, proposed by Kovac *et al.* [29], has a set of four rules:

1) $R > 95 \quad \text{and} \quad G > 40 \quad \text{and} \quad B > 20.$
2) $max(R,G,B) - min(R,G,B) > 15.$
3) $|R - G| > 15.$
4) $R > G \quad \text{and} \quad R > B.$

If the four rules are true, the pixel is classified as "skin", otherwise, it is classified as "background". To facilitate its identification, in this paper we will call this algorithm *Kovac*.

The second algorithm has a very simple rule proposed by Al-Shehri [28], which considers only the value of *R* and *G*. This rule classifies the pixel as "skin" when $R - G$ is greater than 20 and less than 80 ($20 < R - G < 80$), otherwise, the pixel is classified as "background". In this work that algorithm will be identified as *Al-Shehri*.

In the third algorithm, proposed by Osman *et al.* [30], two rules are checked to classify the pixel:

1) $0.0 <= \frac{R-G}{R+G} <= 0.5.$
2) $\frac{B}{R+G} <= 0.5.$

If the two rules are true, the pixel is classified as "skin", otherwise, it is classified as "background". We will identify this algorithm as *Osman* in this work.

Finally, the fourth algorithm, proposed by Swift [31], consists of the following rules:

1) $B > R.$
2) $G < B.$
3) $G > R.$
4) $B < (\frac{1}{4})R.$
5) $B > 200.$

If at least one of the five rules is true, the pixel is classified as "background", otherwise, it is classified as "skin". In this work that algorithm will be identified as *Swift*. The test results of the five implemented segmentation methods will be presented in Section V.

C. Tracking

This stage aims to isolate only the regions of image points belonging to the hands for feature extraction. In order to achieve this objective, we implemented an algorithm to track the hands of the individual in the images sequences and produce new images delimited by the region where they are, without the help of sensors or markers, looking for regions that

[7]http://www.cs.waikato.ac.nz/ml/weka

minimize a distance function taking in account the colors and the distance between the new hand position and the position in the previous (reference) image. Fig. 4 shows an example of result obtained by applying the implemented hand tracker in a pre-processed and segmented images sequence.

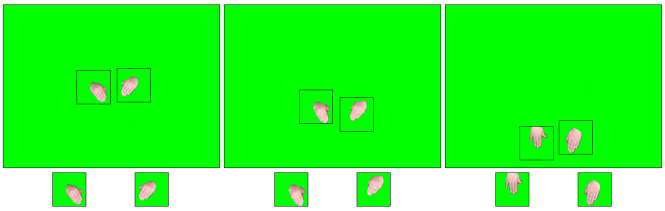

Fig. 4. Results obtained by the hand tracker in a pre-processed and segmented images sequence.

D. Feature extraction

The feature extraction stage consists in extracting relevant properties of the image sequences in order to provide information to recognize the sign. Based on the studies reviewed in this work, the following characteristics of image sequences were extracted: hand shapes, pixel frequency, hand displacements, distance between the hands, and the distance between the hands and the face; amounting to nine feature vectors.

Hand shapes were represented using the extractor proposed by Digiampietri *et al.* [32]. Each hand is represented as a set of 180 points, each of them contains the normalized distance between the image center of mass and the corresponding hand contour point for each other degree (from 0° to 358°). Pixel frequency characteristic is the percentage of pixels belonging to skin color and the background in each image of the images sequence generated by the tracker. For the extraction of other features, while applying the tracker the center of mass of the hands was obtained from each image of the images sequence. From the center of mass were measured the displacements of the hands, the distance between the hands and the distance between the hands and the face through the Euclidean distance between the centers of mass.

To obtain the center of mass of the face, before applying the steps of pre-processing, segmentation and tracking previously defined, an implementation of the method proposed by Viola and Jones [33] for face detection was used. It is present in the OpenCV library and, in this study, achieved 100% success rate to detect the faces of individuals in all images sequences from the image database.

After they were extracted, the values of all features vectors were normalized to a range between 0 and 1. These vectors are used in the recognition as explained in the next section.

E. Recognition

For the recognition and classification of words in BSL, a variation of the Levenshtein distance technique [34] was used. The original Levensthtein distance technique is presented in Equation 1 (lev), where a and b represent characters arrays, i and j represents the indexes of these arrays, and $d(a_i, b_j) = 0$ if $a_i = b_j$; or $d(a_i, b_j) = 1$, otherwise.

$$lev_{a,b}(i,j) = \begin{cases} max(i,j) & \text{if } min(i,j) = 0, \\ min \begin{cases} lev_{a,b}(i-1,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j+1)+d(a_i,b_j) \end{cases} & \text{otherwise} \end{cases} \quad (1)$$

Our variation of the Levenshtein distance technique is presented in Equation 2 ($\widehat{lev}$), where a and b represent features vectors and $\widehat{d}(a_i, b_j) = abs(a_i - b_j)$. This variation allows a non binary comparison of the features.

$$\widehat{lev}_{a,b}(i,j) = \begin{cases} max(i,j) & \text{if } min(i,j) = 0, \\ min \begin{cases} \widehat{lev}_{a,b}(i-1,j)+0.2 \\ \widehat{lev}_{a,b}(i,j-1)+0.2 \\ \widehat{lev}_{a,b}(i-1,j+1)+\widehat{d}(a_i,b_j) \end{cases} & \text{otherwise} \end{cases} \quad (2)$$

At first, the recognition technique calculates the edit distance between the nine features vectors of the word and each of the other samples words of the database. Then it uses a binary classifier that checks for a possible match between the performed word and each word of the database, based in the nine distances calculated. Finally, it verifies what word of the database received more votes by the classifier, which is then presented as the result of the recognition.

In this work, Weka's implementation of the *Random Forest* method [35] was used as classifier. The results of the tests are detailed in the next section.

V. Experimental Results

First, in order to choose the segmenter used to segment the pre-processed images sequences, some tests were made with the five implemented segmentation methods, detailed in Subsection IV-B, using a set of 200 frames taken from the 20 recorded videos to build the image database (10 frames from each video), after the pre-processing step.

The test results were evaluated considering accuracy, sensitivity and specificity measures, in addition to the overlap coefficient. The overlap coefficient is one of the most cited metrics in the literature to measure segmentation, which is the relative area of the intersection of two regions considered [36]. A value of 0 indicates the worst performance, i.e., there is no intersection between the considered correct area and the automatically obtained area. A value of 1, in turn, indicates a perfect segmentation. The results obtained are presented in Table II.

TABLE II
Results obtained by the five tested segmenters in the pre-processed images sequence.

Segmenter	Accuracy	Sensitivity	Specificity	Overlap
[29]	98.86%	70.50%	99.49%	0.59
[28]	98.62%	70.20%	99.27%	0.54
[30]	98.07%	80.86%	98.42%	0.48
[31]	98.04%	54.05%	98.92%	0.34
RF	96.07%	94.84%	96.09%	0.36

Though the *RotationForest (RF)* algorithm had presented the best result to classify the pixels, compared to the other Weka algorithms, analyzing the results in Table II and the images produced, shown in Fig. 5, we realized that the best strategy to segment the frames from our sign database was using the segmenter proposed by Kovat *et al.* [29], which was then chosen to be used in this work.

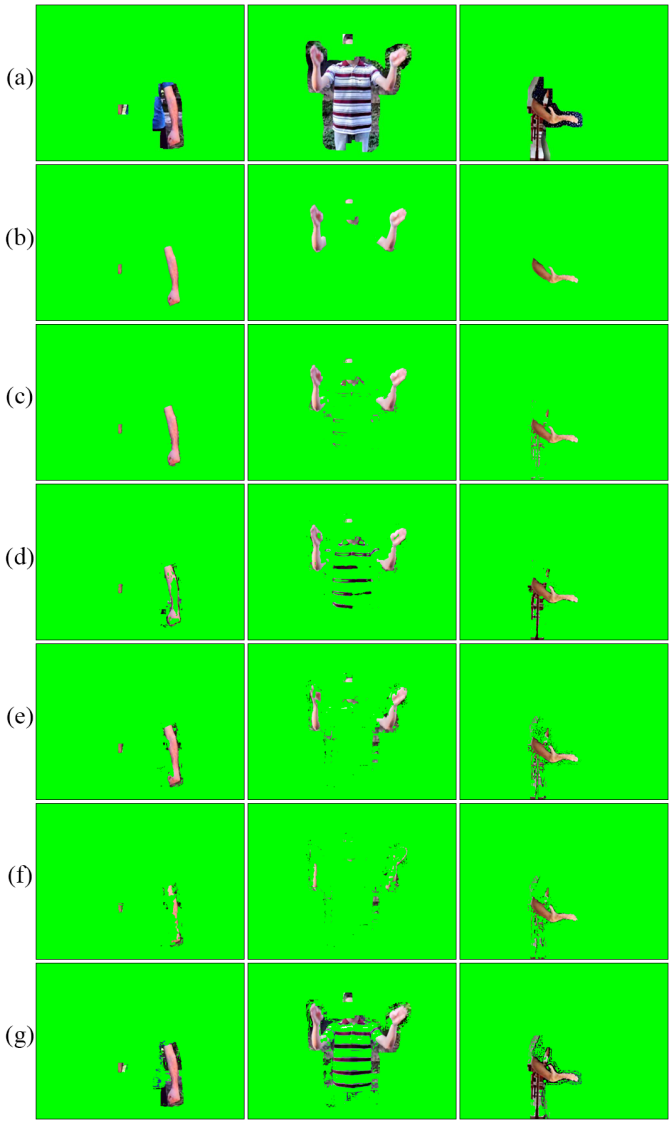

Fig. 5. Examples of results produced by segmentation algorithms implemented: (a) preprocessed frames; (b) frames expected (manual segmentation); (c) *Kovac*; (d) *Al-Shehri*; (e) *Osman*; (f) *Swift*; (g) *RotationForest.*

We notice that the pre-processing techniques used before the segmenter substantially improved the results. Without the pre-processing step, the average rate of accuracy, sensitivity, specificity and the overlap index obtained by the segmenter proposed by Osman *et al.* [29] decreases considerably to 97.03%, 56.59%, 97.87% and 0.29, respectively.

To validate the proposed solution, we constructed a dataset as follows. For each of the image sequences (words) extracted from each of the videos, the Levenshtein distances of its features with the features of other remaining samples words were calculated. From the 20 recorded videos, two videos were not used because the poor results obtained after the segmentation, which prevented the execution of the remaining steps of the system. It is believed that the poor results were obtained by the influence of the clothes that the individuals used during the recording and the environment in which the videos were recorded. With the removal of these two videos, the average rate of accuracy and the overlap index obtained by the segmenter proposed by Osman *et al.* [29] increased considerably to 99.02% and 0.61, respectively.

After the removal of those two videos, from the 18 videos used to test the recognition of words, 422 samples words (image sequence) were extracted. It is worth mentioning that there is only one sample of each of the 30 words in each of the videos, totaling 540 samples. But from these 540 samples, 118 presented problems during the segmentation, which made the tracking stage unfeasible. So, to test the recognition of words, 422 samples were used, with an average of about 14 samples per word. To evaluate the classifier, the 2-fold cross-validation technique was used. The system correctly recognized all the 422 samples, obtaining an impressive 100% accuracy for these 422 signals which were well segmented automatically.

It is therefore worth highlighting that, from the original 600 samples words, the developed system was able to correctly segment 422 (70%). From these 422 words, the recognition system was able to correctly recognize all of them, despite the chosen signs having superpositions of hand configurations and movements.

VI. Conclusion

In this paper, we implemented and evaluated a promising strategy to automatically recognize BSL signs from video sequences, using a combination of digital image processing techniques, the Levenshtein distance technique and a voting scheme with a binary classifier to carry out the recognition.

The proposed system performs the recognition of the signs using only hand configuration, orientation, point of articulation and movement as parameters. Thus, as future work, a possible evolution would be to incorporate non-manual expressions, an important parameter in sign language recognition which, until now, has been little used in the works of sign recognition found in the literature.

We notice that the pre-processing techniques used before the segmenter substantially improved its results. The proposed strategy to segment ROI in the images sequences obtained an average accuracy rate greater than the rates found in reviewed papers [7], [17], which use considerably more complex techniques than those used in this work to segment the images.

It is believed that all stages of the proposed system, including the comparison between segmentation strategies based on machine learning, are contributions for future work in the sign language recognition area. Moreover, as an additional contribution, we expect the image database built, which is available by contacting the authors, can be used by other researchers.

ACKNOWLEDGMENT

We would like to thank the Brazilian Federal Agency for Support and Evaluation of Graduate Education (CAPES) for its financial support for this work.

REFERENCES

[1] J. a. P. S. Neto and L. Oquendo, "Estudo do estado da arte das técnicas de reconhecimento das línguas de sinais por computador," in *Anais do VI Congresso Tecnológico INFOBRASIL TI & TELECOM*, Fortaleza, 2013.

[2] G. F. C. Campos, S. Barbon, and R. G. Mantovani, "A meta-learning approach for recommendation of image segmentation algorithms," in *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, Oct 2016, pp. 370–377.

[3] E. Escobedo and G. Camara, "A new approach for dynamic gesture recognition using skeleton trajectory representation and histograms of cumulative magnitudes," in *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, Oct 2016, pp. 209–216.

[4] N. A. Albres, *Surdos & Inclusão Educacional*. Editora Arara Azul, 2010.

[5] R. M. Quadros and L. B. Karnopp, *Língua de Sinais Brasileira - Estudos linguísticos*. Editora Artmed, 2004.

[6] S. Wilcox and P. P. Wilcox, *Aprender a Ver*. Editora Arara Azul, 2005. [Online]. Available: http://editora-arara-azul.com.br/pdf/livro2.pdf

[7] D. Van Hieu and S. Nitsuwat, "Image preprocessing and trajectory feature extraction based on hidden markov models for sign language recognition," in *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD '08. Ninth ACIS International Conference on*, Aug 2008, pp. 501–506.

[8] P. Goh and E. Holden, "Dynamic fingerspelling recognition using geometric and motion features," in *Image Processing, 2006 IEEE International Conference on*, Oct 2006, pp. 2741–2744.

[9] M. P. Paulraj, S. Yaacob, H. Desa, C. R. Hema, W. M. Ridzuan, and W. A. Majid, "Extraction of head and hand gesture features for recognition of sign language," in *Electronic Design, 2008. ICED 2008. International Conference on*, Dec 2008, pp. 1–6.

[10] H. Hienz, K. Grobel, and G. Offner, "Real-time hand-arm motion analysis using a single video camera," in *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*, Oct 1996, pp. 323–327.

[11] D. Dimov, A. Marinov, and N. Zlateva, "Cbir approach to the recognition of a sign language alphabet," in *Proceedings of the 2007 International Conference on Computer Systems and Technologies*, ser. CompSysTech '07. New York, NY, USA: ACM, 2007, pp. 96:1–96:9. [Online]. Available: http://doi.acm.org/10.1145/1330598.1330700

[12] V. Radha and M. Krishnaveni, "Threshold based segmentation using median filter for sign language recognition system," in *Nature Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, Dec 2009, pp. 1394–1399.

[13] N. Habili, C. C. Lim, and A. Moini, "Segmentation of the face and hands in sign language video sequences using color and motion cues," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 14, no. 8, pp. 1086–1097, Aug 2004.

[14] H. L. Ribeiro and A. Gonzaga, "Hand image segmentation in video sequence by gmm: A comparative analysis," in *Computer Graphics and Image Processing, 2006. SIBGRAPI '06. 19th Brazilian Symposium on*, Oct 2006, pp. 357–364.

[15] D. Mushfieldt, M. Ghaziasgar, and J. Connan, "Robust facial expression recognition in the presence of rotation and partial occlusion," in *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, ser. SAICSIT '13. New York, NY, USA: ACM, 2013, pp. 186–193. [Online]. Available: http://doi.acm.org/10.1145/2513456.2513493

[16] T. Kim, G. Shakhnarovich, and K. Livescu, "Fingerspelling recognition with semi-markov conditional random fields," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, Dec 2013, pp. 1521–1528.

[17] J. Han, G. Awad, and A. Sutherland, "Automatic skin segmentation and tracking in sign language recognition," *Computer Vision, IET*, vol. 3, no. 1, pp. 24–35, March 2009.

[18] M. El-Jaber, K. Assaleh, and T. Shanableh, "Enhanced user-dependent recognition of arabic sign language via disparity images," in *Mechatronics and its Applications (ISMA), 2010 7th International Symposium on*, April 2010, pp. 1–4.

[19] V. M. Gonçalves, S. M. Peres, R. A. P. Oliveira, and M. E. Delamaro, "Desempenho de funções de similaridade em cbir no contexto de teste de software: Um estudo de caso em segmentação de imagens de gestos de libras," in *Proceedings of the VIII Workshop de Visão Computacional (WVC)*. Brazilian Computer Society, 2012, pp. 1–6.

[20] N. Soontranon, S. Aramvith, and T. H. Chalidabhongse, "Face and hands localization and tracking for sign language recognition," in *Communications and Information Technology, 2004. ISCIT 2004. IEEE International Symposium on*, vol. 2, Oct 2004, pp. 1246–1251 vol.2.

[21] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," in *Computer Vision, 1995. Proceedings., International Symposium on*, Nov 1995, pp. 265–270.

[22] O. Marques Filho and H. Vieira Neto, *Processamento Digital de Imagens*. Rio de Janeiro, Brazil: Editora Brasport, 1999.

[23] K. Zuiderveld, "Contrast limited adaptive histogram equalization," in *Graphics Gems IV*, P. S. Heckbert, Ed. San Diego, CA, USA: Academic Press Professional, Inc., 1994, pp. 474–485. [Online]. Available: http://dl.acm.org/citation.cfm?id=180895.180940

[24] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 2 - Volume 02*, ser. ICPR '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 28–31. [Online]. Available: http://dx.doi.org/10.1109/ICPR.2004.479

[25] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recogn. Lett.*, vol. 27, no. 7, pp. 773–780, May 2006. [Online]. Available: http://dx.doi.org/10.1016/j.patrec.2005.11.005

[26] R. C. Gonzalez and R. E. Woods, *Processamento de Imagens Digitais*. Edgard Blucher, 2000.

[27] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," *SIGKDD Explorations*, vol. 11, pp. 10–18., 2009.

[28] S. A. Al-Shehri, "A simple and novel method for skin detection and face locating and tracking," in *Computer Human Interaction, 6th Asia Pacific Conference, APCHI 2004, Rotorua, New Zealand, June 29 - July 2, 2004, Proceedings*, 2004, pp. 1–8. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-27795-8_1

[29] J. Kovac, P. Peer, and F. Solina, "Human skin color clustering for face detection," in *EUROCON 2003. Computer as a Tool. The IEEE Region 8*, vol. 2, 2003, pp. 144–148 vol.2.

[30] G. Osman, M. S. Hitam, and M. N. Ismail, "Enhanced skin colour classifier using RGB ratio model," *CoRR*, vol. abs/1212.2692, 2012. [Online]. Available: http://arxiv.org/abs/1212.2692

[31] D. Swift, "Evaluating graphic image files for objectionable content," May 17 2005, uS Patent 6,895,111. [Online]. Available: http://www.google.com/patents/US6895111

[32] L. Digiampietri, B. Teodoro, C. Santiago, G. Oliveira, and J. Araújo, "Um sistema de informação extensível para o reconhecimento automático de libras," in *SBSI 2012 - Trilhas Técnicas (Technical Tracks)*, São Paulo, SP, Brazil, 2012.

[33] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition*, 2001, pp. 511–518.

[34] V. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Physics Doklady*, vol. 10, p. 707, 1966.

[35] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. [Online]. Available: http://dx.doi.org/10.1023/A%3A1010933404324

[36] N. P. Gruszauskas, K. Drukker, M. L. Giger, C. A. Sennett, and L. L. Pesce, "Performance of breast ultrasound computer-aided diagnosis: dependence on image selection." *Acad. Radiol*, vol. 15(10), pp. 1234–1245, 2008.