Computación y Sistemas
ISSN: 1405-5546
computacion-y-sistemas@cic.ipn.mx
Instituto Politécnico Nacional
México

Ben Jmaa, Ahmed; Mahdi, Walid; Ben Jemaa, Yousra; Ben Hamadou, Abdelmajid

A New Approach For Hand Gestures Recognition Based on Depth Map Captured by RGB -D Camera

Computación y Sistemas, vol. 20, núm. 4, 2016, pp. 709-721
Instituto Politécnico Nacional
Distrito Federal, México

Available in: http://www.redalyc.org/articulo.oa?id=61549258011

ISSN 2007-9737

A New Approach For Hand Gestures Recognition Based on Depth Map Captured by RGB-D Camera

Ahmed Ben Jmaa[1], Walid Mahdi[1], Yousra Ben Jemaa[2], Abdelmajid Ben Hamadou[1]

[1] Multimedia, Information systems and Advanced Computing Laboratory, Sfax, Tunisia

[2] Signal and System Research Unit, Tunis, Tunisia

ahmed.benjmaa@gmail.com, walid.mahdi@isimsf.rnu.tn, yousra.benjemaa@enis.rnu.tn, abdelmajid.benhamadou@isimsf.rnu.tn

Abstract. This paper introduces a new approach for hand gesture recognition based on depth Map captured by an RGB-D Kinect camera. Although this camera provides two types of information "Depth Map" and "RGB Image", only the depth data information is used to analyze and recognize the hand gestures. Given the complexity of this task, a new method based on edge detection is proposed to eliminate the noise and segment the hand. Moreover, new descriptors are introduce to model the hand gesture. These features are invariant to scale, rotation and translation. Our approach is applied on French sign language alphabet to show its effectiveness and evaluate the robustness of the proposed descriptors. The experimental results clearly show that the proposed system is very satisfactory as it to recognizes the French alphabet sign with an accuracy of more than 93%. Our approach is also applied to a public dataset in order to be compared in the existing studies. The results prove that our system can outperform previous methods using the same dataset.

Keywords. Sign Language (SL), Kinect camera, Depth sensor, Hand gesture recognition.

1 Introduction

Sign Language is at the same time a very promising and challenging field of study given the big number of hearing impaired people who cannot express their needs and communicate with others. To avoid this problem and its associated societal and economic impact, intelligent systems have been proposed.

Hand gesture recognition is one of the most important tasks in the sign language research area. Since it is a difficult field suffering from complex problems related to the study context, it needs the use of robust descriptors.

Many approaches have been developed in literature to achieve this challenging objective and perform a gesture recognition using a simple camera [27, 4, 22].

Although these approaches are intuitive, easy and cheap, the results are still not sufficient especially for real-time applications. This may be due to the limitation of the optical sensor in the given quality of the captured image which is sensitive to lighting conditions and complex backgrounds.

Using a depth sensor leads to better results [23, 28, 18]. Approaches using this sensor can be classified into two categories: static and dynamic. Static methods treat static gestures that are represented by individual movements and then by a single image; however, the dynamic approaches deal with dynamic gestures represented by image sequences.

This paper focuses only on the static hand gestures. It also proposes and evaluates a new approach for the recognition of hand gestures of the 23 static letters of the French alphabet.

In vision-based hand gesture recognition, the key factor is the accurate and fast hand tracking and segmentation. This challenge is very difficult to achieve due to the complex backgrounds and illumination variations especially when a standard camera is used.

Our system aims at achieving fast and accurate hand gesture recognition based on the depth Map captured by a Kinect camera.

Many steps are performed to achieve our proposed system as illustrated in Figure 1.

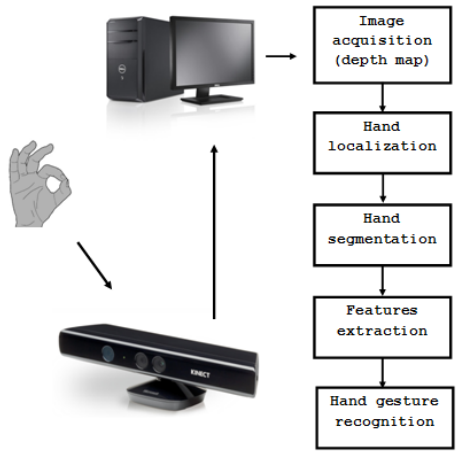


Fig. 1. The proposed recognition system

The remaining part of this paper is organized as follows: Section two presentes the related studies about hand gesture recognition before describing the hand localization process in section three. Section four, however, introduces a new technique of image segmentation to remove the complex background and noise from the hand image. In section five different new descriptors for hand gesture recognition are proposed. These are classified in two categories : 2D descriptors and 3D descriptors extracted from depth information. Section six, describes the performed tests on two datasets, by evaluating the relevance of the proposed system and comparing it to the previous studies in terms of classification rate. The paper conclusion as well as some potential future studies are presented in the last section.

2 Related Works

Several methods have been developed for hand gesture recognition. These can be classified into many categories. The first one is based on glove-analysis [13], used to control a virtual actor, describe and manipulate objects on computer screens [1, 2] or even recognize the sign language [5].

Although these approaches are not complex and intuitive, they still remain expensive.

The second category is the vision-based analysis, which relies on hand gesture acquisition with camera for Human computer interaction (HCI) application or sign language recognition. These methods need to localize the hand in an image captured by a simple webcam or a Microsoft Kinect camera.

In [4], the authors developed a drawing application that provides a real-time position of a finger, by correlation to the fingertip.

In [11], they proposed deformable models by using a point-distribution model, which represents any form of the skeleton by a set of feature points and variation patterns that describe the movements of these points. This model is made from a set of training sequences by a singular value decomposition of the differences between the forms of a set of training sequence and the averaged form. Recognition is based on the model of distribution points, proposed by Martin and Crowley [22].

The system developed in [23] employs a combined RGB and depth descriptor in order to classify the hand gestures. Two interconnected modules are employed: the first detects a hand in the region of interaction and performs a user classification, and the second performs the gesture recognition.

A robust hand gesture recognition system using the Kinect sensor is built in [28]. The authors also proposed a novel distance metric for a hand dissimilarity measure, called Finger-Earth Mover's Distance (FEMD).

In [18], the authors proposed a highly precise method to recognize static gestures from depth data. A multi-layer random forest (MLRF) is then

trained to classify the feature vectors, which leads to the recognition of the hand signs.

In [14], the authors have proposed a new method for the American Sign Language alphabet (ASL) recognition using a low-cost depth camera. This new method uses a depth contrast feature based on per-pixel classification algorithm to segment the hand configuration. The authors used specific gloves to ensure the hand segmentation step. Then, a hierarchical mode-seeking method is developed and implemented to localize the hand joint positions under kinematic constraints. Finally, a Random Forest (RF) classifier is built to recognize the ASL signs using the joint angles.

In [20], the writers proposed a novel method for contact-less HGR using Microsoft Kinect for Xbox. Their system is can detect the presence of gestures, identify fingers, and recognize the meanings of nine gestures in a pre-defined Popular Gesture scenario.

In [30], a new superpixel-based hand gesture recognition system is proposed. This system is based on a novel superpixel earth mover's distance metric, together with Kinect depth camera, to measure the dissimilarity between the hand gestures. This measurement is not only robust to distortion and articulation, but also invariant to scaling, translation and rotation with a proper preprocessing.

In [3], the authors proposed a new approach for digit recognition based on a hand gesture analysis from RGB images. They extracted and combined a set of features modeling the hand shape with an induction graph. Their approach is invariant to scale, rotation and translation of the hand.

3 Hand Localization

Hand detection is a very important step to start the hand gesture recognition process. In fact, to identify a gesture, it is necessary to localize the hands and their characteristics in the image. We used a Kinect sensor presented in Figure 2 as an input device which captures the color image and the depth map.

The RGB image can be displayed as in all the cameras. The Kinect sensor produces 640x480 images when using 30 fps and 1280x1024 at 15 frames per second.

The Kinect can capture depth information by projecting an infrared dot pattern and its subsequent capture by an infrared camera [7]. The depth information can be extremely interesting to detect shapes on the Kinect's video stream. This feature, would be used in our context to localize and get the hand shape while discarding other information.

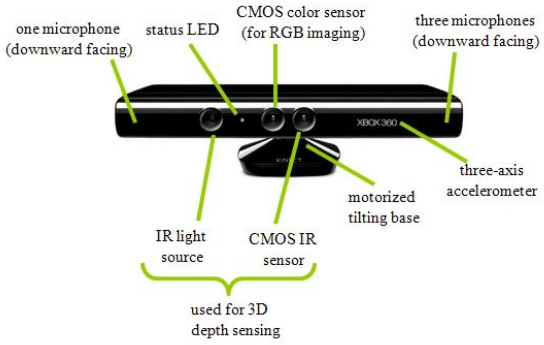


Fig. 2. Microsoft Kinect sensor

3.1 Depth Map

The depth information is the determinant factor in our application. It indeed gives each pixel depth for the sensor. Thus, in addition to the 2D position of each pixel and its color, we also have its depth. This makes it far to search for the hands.

First of all, the depth information is converted into a gray scale image.

The key point here is therefore the Kinect's ability to give us three-dimensional information.

When the Kinect follows a person precisely, it can provide a skeleton from all the key points already detected.

To achieve better results, it is necessary that the person stand at a distance of 1.2 to 3.5 meters from the camera [7]. Above these limits, the accuracy of the sensor decreases rapidly and is therefore not possible to follow the person.

As shown in Figure 3, there are 20 key points (which will be called joints) that are detected and tracked.

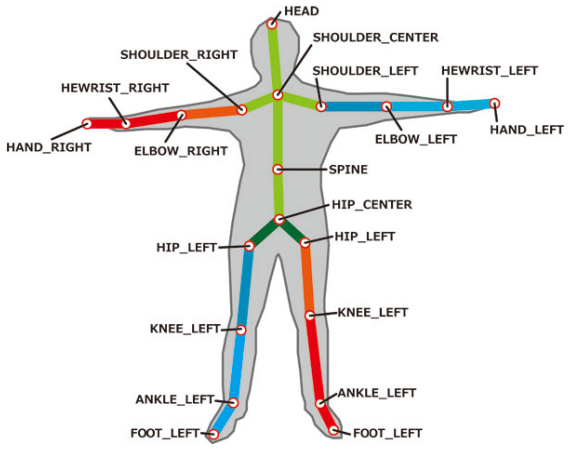


Fig. 3. Kinect skeleton

The depth data are stored in the form of integer arrays of 16 bits. The depth information can be recovered in 320x240 or in 80x60.

The 16 bits of each pixel are as follows:

— The 13 high order bits give the distance from the camera in millimeters for each pixel.

— The 3 lower bits give the index of this person. This index is 0 if the Kinect has effectively detected a person.

3.2 Hand Detection from Depth Map

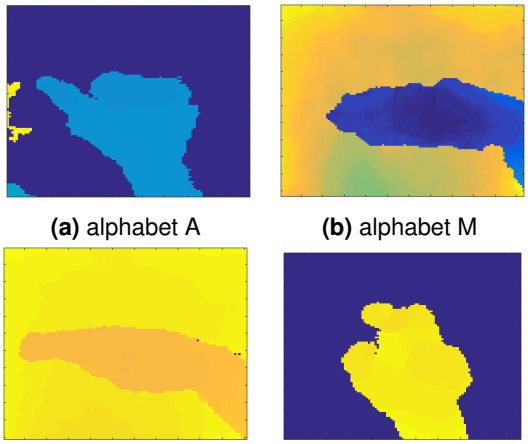
In our application that aims to analyze and recognize the French alphabet, we are interested only in the right hand.

Consequently, from the matrix containing the 20 key point coordinates, we get only point number 12 since it corresponds to the center of the right hand palm. The coordinate of this point is used to localize the hand position in the image.

4 Hand Segmentation

The hand segmentation step follows the hand localization to remove all the useless and annoying information like noise and background.

In Figure 4, we present the output images after the detection step from the depth map. It should be noted that these images can contain beside the hand, noise, background and other parts of the body.



(a) alphabet A **(b)** alphabet M

(c) alphabet N **(d)** alphabet X

Fig. 4. Hand detection: (A) Hand with noise, (M) Hand with body and background, (N) Hand with body and (X) Only the hand

Many approaches have been proposed in the literature to segment a hand and remove all the other regions in images. Methods based on thresholding, like that of Otsu [21] are not robust. We, then, suggest a new approach based on the edge detection considered as the most common method for detecting significant discontinuities in an image [15].

In what follows, we introduce our new approach for hand segmentation based on edge detection. Our segmentation approach steps can be stated as follows:

— Noise elimination by applying the bilateral filter on the image [9, 10, 16].

— Edge detection using the ”Canny” method [8].

— Closing the edge regions.

— Filling the hand region.

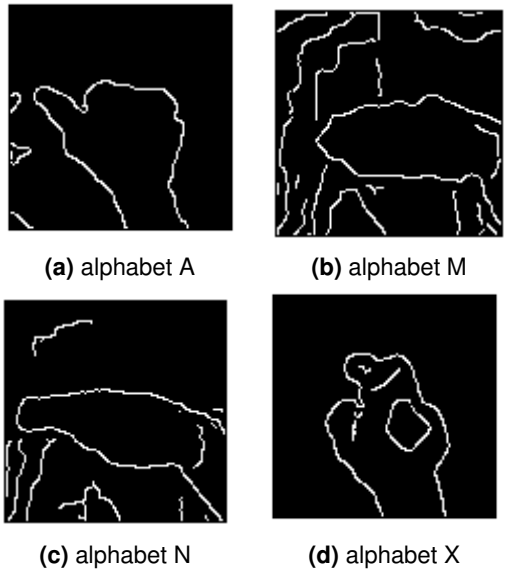
— Removing all the unwanted edge regions.

ISSN 2007-9737

4.1 Edge Detection

Many edge detection methods, such as Sobel filtering, Perwit filtering and Canny operator have been proposed in the literature. The most powerful one is the Canny method [8], because it uses two different thresholds to detect both strong and weak edges. The weak edges are included only if they are connected to strong ones.

In Figure 5, we show the edge detection results using the Canny operator applied on already localized hands. It should be noted that the obtained results are not very satisfactory and many erroneous edges are detected. This might be due to the presence of noise.



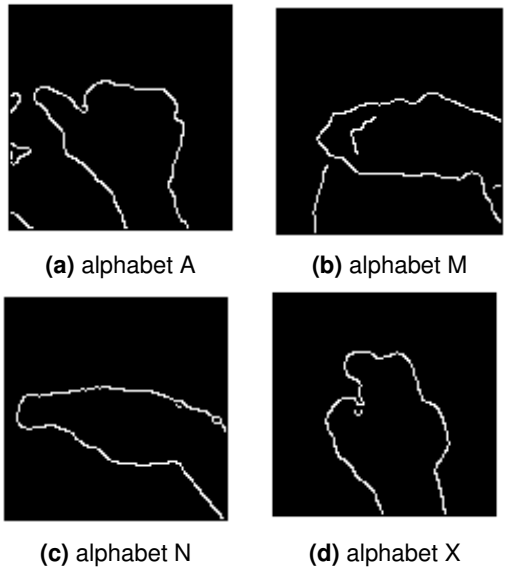
(a) alphabet A **(b)** alphabet M

(c) alphabet N **(d)** alphabet X

Fig. 5. Edge detection with canny operator

In order to improve the edge detection results, we propose to eliminate the noise before the canny detection step by applying a bilateral filter which would preserve the edge and reduce the noise [9, 10, 16].

Figure 6 shows the effectiveness of the bilateral filter in reducing the noise in the image and going better results after the edge detection with the Canny method.



(a) alphabet A **(b)** alphabet M

(c) alphabet N **(d)** alphabet X

Fig. 6. Edge detection with canny operator after applying the bilateral filter on the localized hand image

4.2 Edge Closing

After the edge detection step, many imprecise results, like incomplete or open contours are found. A step of contour closing is therefore necessary.

Several researchers have only used morphological operations, such as dilation and erosion to close the edges. In this paper, we introduce a new approach for closing the edges relying on both the morphology operator and many other forms of control in order to improve the results and make sure that the hand's edge is closed.

The new approach of the edge closing is presented in Algorithm 1. Figure 7 represents the hand's regions after an edge closing step using Algorithm 1.

4.3 Filling the Hand Region

After closing the contours of the image, two cases can be faced:

— a single-region image,

— a several-region image.

Algorithm 1: Closing hand edge

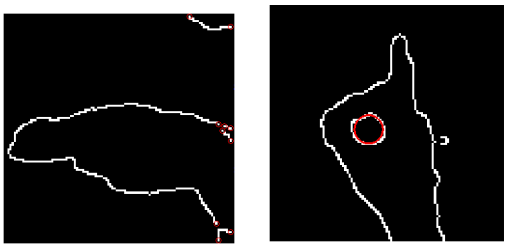
```
// Edge Image and the depth map
input      : ImEdge, ImDepth
output     : ImEdge

// Find the minimum value of pixels in the
   depth map (We assume that the hand is the
   closest region to the camera)

minDepth ← Minimum(ImDepth);

// Find the coordinate of the pixels having
   the minimum value of pixels in the depth
   map
[minPixels] ← Find(minDepth);

repeat// rBool==1 Or hBool==1
    // See Algo 2
    rBool ← VerificationEdgeClosure(ImEdge);
    if Regions are not closed then // rBool==0
        // Delete found closed circle region
           and save it in an array.  This
           step is very important because of
           the many iteration of applying
           morphological operators like
           dilation and erosion can fill
           these regions.  The circle region
           is important and used for some
           features extraction which will be
           described in section 5
        // See Algo 3
        [CircleArray] ← FindCircleArea(ImEdge);
        // See Algo 4
        hBool ← HandClosure(ImEdge,minPixels);
        if Hand region is not closed then // hBool
        == 0
            // In this step of the while
               loop, the hand region or all
               the regions are not closed, we
               apply the morphology operators
               such as dilation and erosion,
               to add pixels to the existant
               edge with "disk" structural
               element
            [ImEdge] ← Dilate(ImEdge);
            [ImEdge] ← Erode(ImEdge);
        end
    end
until hand region is closed;
// Restore the circular regions
ImEdge ← ImEdge + CircleArray;
Return(ImEdge);
```



(a) End points **(b)** Circle region

Fig. 7. Edge closing using the proposed method

Algorithm 2: Verification of edge closure

```
input      : ImEdge
output     : Boolean

// We use a temporary edge image for
   thinning because of the many iterations
   of thinning can change the shape of the
   regions

[ImTemp] ← [ImEdge];
// Thin edges to lines [19]

[ImTemp] ← Thins(ImTemp);

// Find the end points in the thinned edge
   image

[endPoints] ← FindEndpoints (ImTemp);
if End points do not exist then
// Size(endPoints)==0
    // All the regions are closed
    Return(1);
end
Return(0);
```

Since the hand is the closest object to the camera, the pixels having the minimum depth value belong or are very close to the hand region. Consequently, even if there are several regions, the desired one namely the hand region is defined as the area that includes the minimum depth pixel.

We refer to the work proposed in [29] in order to fill the hand region. Since this work uses only one pixel belonging to the desired area to fill the whole region, we have to find only one pixel from the hand region.

Therefore, we start by looking for the list of pixels with the minimum depth value. For each pixel three cases are possible:

Algorithm 3: Find the circle region

```
input  : ImEdge, circleArray
output : ImEdge, circleArray

[objEdge] ← FindObject(ImEdge);
for i ← 1 to Size(objEdge) do // For each
object
    [ImTemp] ← [objEdge[i];
    [ImTemp] ← Thins(ImTemp);
    [endPoints] ← FindEndpoints(ImTemp);
    if End points not exist then
    // size(endPoints)==0
        // Find circles using circular Hough
           transform
        [circleObj] ← Findcircle(objEdge[i]);
        if Object has a circular shape then
        // size(circleObj) != 0
            [circleArray] ← circleObj;
            [ImEdge] ← ImEdge - ImTemp;
        end
    end
end
Return(ImEdge, circleArray,);
```

Algorithm 4: Verify the hand regions closure

```
input  : ImEdge, minPixels
output : Boolean

// Find the nearest edge of the minPixels,
   the edge represents the hand contours
[objEdge] ← FindObject(ImEdge);
for i ← 1 to Size(objEdge) do // For each
object
    [distObj] ←
    distEuclidean(minPixels, objEdge[i]);
end
// Get the list of objects having the
   minimum distance with minPixels
[handObj] ← Minimum(distObj);
if Only one object has the minimum distance then
// Size(handObj)==1
    [ImTemp] ← [handObj];
    [ImTemp] ← Thins(ImTemp);
    [endPoints] ← FindEndpoints(ImTemp);
    if End points do not exist then Return(1);
    ;                        // Size(endPoints)==0
    // Hand region is closed
    else Return(0);
    ;  // Ambiguity (Several/Nor objects have
    the same minimum distance)
end
Return(0);
```

— Pixel on the hand contours: In this case pixel value=1; so, we can not fill the region [29].

— Pixel outside the hand region: Since this pixel is characterized by a minimum depth value, it is necessarily located near the contour of the hand.

— Pixel in the hand region: This pixel can be far from the contour or very close to the hand edge.

Since the pixel that has to be chosen to the fill the region step must be in the hand region and far from the contours [29], it must not have a neighborhood belonging to the contour image. We propose to apply 5*5 windows for all the pixels (having the minimum depth) to verify all the neighborhood values. The chosen pixel must have all neighborhood values equal to zero.

When a filling pixel inside the hand region is not found (all pixels are on or close to the contour region), we increment the used minimum depth value over a millimeter to have another list of pixels, and then we repeat the same search process described above.

The obtained binary image is considered as a mask to be applied on the depth image and remove all the other pixels as illustrated in Figure 8.

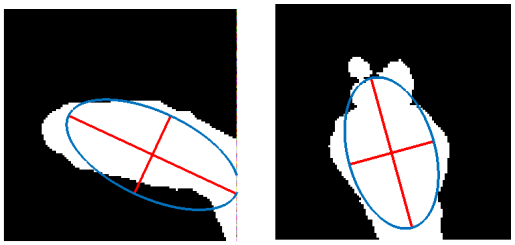
5 Feature Extraction

5.1 Proposed Descriptors

The proposed recognition system is based on two types of features: the first consists of 2D features representing the hand deformation in the 2D plan and describing the geometry information and the hand structure to differentiate between the hand shapes. The second, however, represents the depth information (3D features) mainly selected to represent the finger positions and palm closure.

(a) alphabet M **(b)** alphabet X

Fig. 8. The resulting image after segmentation process



(a) alphabet M **(b)** alphabet X

Fig. 9. Ellipse representation for hand orientation

The 2D features (hand orientation and euler number) are used by [24] but the hand dimension is modified in our approach compared to the literature to make it invariant to scale. The hand occupancy is invented in our approach.

The 3D features (the variance depth values, depth average related to the minimum value and depth average related to the centroid of the hand) are invented in our approach and have never been used in any other research.

5.2 2D Features

5.2.1 Hand Orientation

The hand orientation changes with the change of the gestures. Therefore, it can be used as a characteristic feature of the hand gesture. Since the human hand has an elliptic shape, we propose here to find the best ellipse that includes the hand. The input parameters are a set of (x, y) points (the x, y coordinates of all the hand regions), and the output is the minor and major axis, while the θ is the orientation for the best ellipse (the one that contains the majority of (x, y) input points). The former parameter is assumed as the hand orientation [3]. Figure 9 gives an illustration of the ellipse including the hand region and the effectiveness of the feature in differentiating between two similar hand gestures from different classes. The θ orientation is given by the angle between the ellipse major axis and the x coordinate axis.

The hand orientation is considered as a gesture hand descriptor also used for hand adjustment step into the vertical direction (normalized direction). This is very important to get all the extracted features invariant to rotation.

5.2.2 Hand Dimension and Occupancy

To estimate the hand dimensions, we try to bind all the hand pixels in the rectangular area having the smallest perimeter, as presented in Figure 10.

(a) alphabet M **(b)** alphabet X

Fig. 10. Rectangle representation for hand dimension

If (l, w) are respectively the length and width of the rectangle, two hand descriptors can be proposed. The first one noted D_{dim} is given by equation 1:

$$D_{dim} = \frac{l}{w}. \tag{1}$$

The second one noted $D_{occupancy}$ is defined by equation 2 as follows:

$$D_{occupancy} = \frac{N}{l.w}, \tag{2}$$

where N is the number of all the hand pixels (white pixels).

These two features are invariant to scale and orientation.

5.2.3 Euler Number

The Euler number can describe the structure of the hand and give its characteristic topological. The Euler number E represents the total number of objects in the image (in our case there is only the hand object) minus the total number of holes in those objects H [17, 25]:

$$E = 1 - H. \tag{3}$$

5.3 Features Based on Depth Information: 3D Features

5.3.1 The Variance Depth Value

The variance depth value V_{depth} is obtained by the difference between the maximum value of depth $depth_{max}$ and the minimum $depth_{min}$:

$$V_{depth} = depth_{max} - depth_{min}. \tag{4}$$

The utility of the global feature is to know the palm closure and hand orientation from Z axis.

5.3.2 Depth Average

We compute two features based on depth average, the first one Avg_c is related to centroid depth information and the second one Avg_{min} is related to the minimum depth information. They are defined in equations 5 and 6:

$$Avg_c = \frac{\sum_i H_{depth}(X_i, Y_j)}{N} - H_{depth}(X_c, Y_c), \tag{5}$$

where $H_{depth}(X, Y)$ represents the depth of the hand pixel with coordinates (X, Y), (X_c, Y_c) are the coordinates of the centroid and N the total number of pixels (X_i, Y_j) in the hand region:

$$Avg_{min} = \frac{\sum H_{depth}(X_i, Y_j)}{N} - H_{depth}(X_{min}, Y_{min}), \tag{6}$$

where $H_{depth}(X, Y)$ represents the depth of the hand pixel with coordinates (X, Y), (X_{min}, Y_{min}) are the coordinate of the pixel having the minimum depth value and N the total number of pixels (X_i, Y_j) in the hand region.

6 Experimental Results

6.1 Evaluation on our Dataset

In this section, we evaluate our approach and all the proposed descriptors by creating a new dataset based on the French sign language alphabet which contains 2300 gesture images from the 23 static letters of the French alphabet. Figure 11 shows the French sign language alphabet.

These images, which contain only the depth information of hand gesture, are captured in different backgrounds and angle views.

This dataset is divided randomly ten times into a set of training images (50%) and a set of test images (the remaining 50%). In the literature, there are several techniques of supervised learning. We evaluate our system with the random forest techniques[6].

The classification accuracy is used in order to evaluate the performance of our system and the proposed descriptors.

6.1.1 Comparative Study between Descriptors

To compare the representative power of the proposed descriptors, we test the performance of each one in terms of classification accuracy as represented in Figure 12.

The results show that the dimension descriptor, which is a 2D-type feature, and the variance depth descriptor which is a 3D-type feature present the highest classification accuracy.

We note that the 3D descriptor based on depth information ranks second, which proves that it represents a very suitable feature in terms of gesture classification. This shows the importance of using the Kinect camera and depth information in our study.

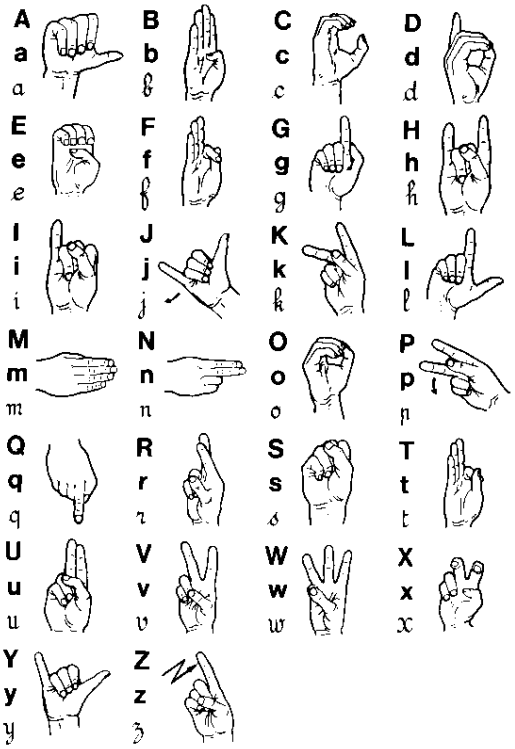


Fig. 11. Alphabet of French sign language

Table 1. Importance of depth features for classification

Descriptor	Accuracy rate
Geometry, 2D	77.8
All (2D-3D)	93.7

6.1.2 Performance of the Proposed System

In order to achieve better accuracy, we take advantage of two types of features to design a new one. The proposed descriptor combines all the proposed descriptors by means of superposition.

Classification accuracy values presented in Table 1 confirm this result. In fact, when using 2D descriptors only, the classification accuracy reaches 77.8%. However, it climbs to (93.7%) when combining 3D and 2D descriptors.

In Table 2, we present the precision rate and the recall rate for the 23 different gestures in the French sign language alphabet.

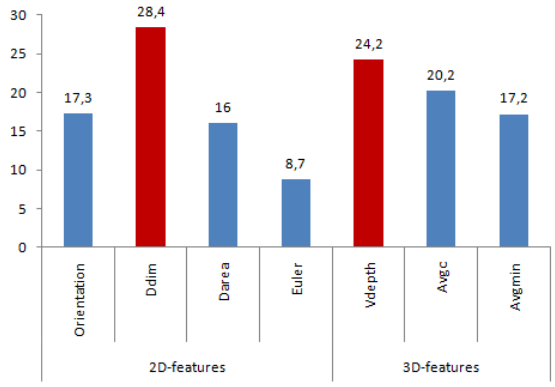


Fig. 12. Accuracy classification for each descriptor

Table 2. Performance of the proposed system

Class	Precision	Recall	Class	Precision	Recall
A	99.3	97.0	N	97.7	97.3
B	90.9	96.0	O	99.0	98.3
C	96.8	99.7	Q	97.6	96.0
D	99.0	97.0	R	92.8	90.7
E	94.6	92.7	S	90.0	93.3
F	93.8	90.7	T	89.9	91.7
G	97.4	98.0	U	94.2	87.0
H	93.3	92.3	V	88.8	84.7
I	77.7	87.0	W	85.9	81.0
K	98.6	96.7	X	88.5	94.7
L	99.3	98.3	Y	98.3	96.0
M	93.7	98.3	-	-	-

6.2 Evaluation on a Public Dataset and Comparison with Existing Work

In order to validate our approach, a comparative study with previous methods on a public dataset is performed.

Our approach is compared to [26] who developed an application to recognize different hand gestures representing the alphabet of the American sign language. This work uses the Kinect camera for hand detection and tracking. To recognize the different gestures, [26] used descriptors based on the Gabor filter [12].

We use the database provided by [26] which contains 500 samples for each of the 24 signs (American alphabet without ”J” and ”Z”) recorded

by 5 different people. The dataset has two types of images, ”RGB” and ”Depth”, captured in different backgrounds and view angles.

Since our approach is based on depth image, we use only depth images (60.000 images) from the dataset to evaluate our system and compare it to the work proposed in [26] which uses the same database (Surrey University dataset), the same protocol test (the 50% of data are used for training and the rest for testing) and the same classifier (the random forest [6]). Proposed descriptors in this work are based on Gabor filters.

Our system can reach 76% of accuracy rate, while [26] attained only 69%. These results shown in table 3 highlight the robustness of our proposed approach, specially, the hand segmentation and the feature extraction steps.

Table 3. Comparison between our approach and that of [26]

Approach	Accuracy rate
Our	76%
[26]	69%

Table 4 shows the precision rate for each sign and for the two approaches.

Table 4. Precision rate of our approach and that of [26] for each American alphabet sign language

Class	Our	[26]	Class	Our	[26]
A	81.8	76.0	N	64.2	66.0
B	88.7	91.0	O	73.4	42.0
C	89.9	67.0	P	75.5	66.0
D	81.1	80.0	Q	83.6	65.0
E	71.2	88.0	R	71.3	48.0
F	78.6	62.5	S	64.1	59.0
G	85.2	65.5	T	60.2	35.0
H	88.1	65.0	U	66.5	86.0
I	76.2	60.0	V	67.0	66.0
K	73.6	64.0	W	62.8	81.0
L	91.2	98.0	X	76.5	68.0
M	61.4	58.0	Y	89.9	90.5

7 Conclusion and Future Works

In this paper, we presented a new approach of hand gesture recognition based on a depth map captured by a Kinect camera. Our approach requires many adjustments and removal steps to segment the hand and avoid noisy information. On the other hand, we proposed new descriptors based on the depth information and showed their relevance to our proposed system. This reflects the importance of using a Microsoft camera Kinect which provides depth information.

Our experimental results show that the approach is effective as its performance reached over 93% when applied on the alphabet French sign dataset. A comparative study enabled us to show that our approach outperforms the existing ones in the state of the art.

Our future focus will be on the use of ”RGB Image” captured by the Kinect camera in the segmentation step to improve the result of hand gesture recognition. We also think about making a 3D-model that will be used in the hand tracking in order to recognize dynamic gestures.

References

1. **Baudel, T. & Beaudouin-Lafon, M. (1993).** Charade: Remote control of objects using free-hand gestures. *Communications of the ACM*, Vol. 36, No. 7, pp. 28–35.
2. **Bellik, Y. (1996).** Modality integration: Speech and gesture. *Survey of the State of the Art in Human Language Technology, Section 9.4*.
3. **Ben Jmaa, A., Mehdi, W., Ben Jemaa, Y., & Ben Hamadou, A. (2009).** Hand localization and fingers features extraction: Application to digit recognition in sign language. *Intelligent Data Engineering and Automated Learning - IDEAL*, pp. 151–159.
4. **Berard, F., Coutaz, J., & Crowley, J. L. (1995).** Finger tracking as input device for augmented reality. *Proc. Intel Workshop on Automatic Face and Gesture-Recognition*, Zurich, Switzerland.
5. **Braffort, A. (1996).** *Reconnaissance et Compréhension de gestes, application à la langue des signes*. Ph.D. thesis, Thèse de l’université de Paris XI, spécialité informatique.

6. **Breiman, L. (2001).** Random forests. *Machine Learning*, Vol. 45, pp. 5–32.
7. **Cai, Z., mad Li Liu, J. H., & Shao, L. (2016).** RGB-D datasets using microsoft Kinect or similar sensors: a survey. *Multimedia Tools and Applications*, volume 75, pp. 1–43.
8. **Canny, J. (1986).** A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 679–698.
9. **Chaudhury, K., Sage, D., & Unser, M. (2011).** Fast o(1) bilateral filtering using trigonometric range kernels. *IEEE Transactions on Image Processing*, Vol. 20, pp. 3376–3382.
10. **Chaudhury, K. N. (2013).** Acceleration of the shiftable o(1) algorithm for bilateral filtering and non-local means. *IEEE Transactions on Image Processing*, Vol. 22, pp. 1291–1300.
11. **Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995).** Active shape models: their training and application. *Computer Vision and Image Understanding*, Vol. 61, No. 1, pp. 38–59.
12. **Daugman, J. (1985).** Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by twodimensional visual cortical filters. *Journal of the Optical Society of America*, pp. 1160–1169.
13. **Dipietro, L., Sabatini, A. M., & Dario, P. (2003).** Evaluation of an instrumented glove for hand-movement acquisition. *Journal of Rehabilitation Research and Development*, Vol. 40, No. 2, pp. 179–190.
14. **Dong, C., Leu, M., & Yin, Z. (2015).** American sign language alphabet recognition using microsoft Kinect. *Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Conference*, pp. 44–52.
15. **Efford, N. (2000).** Digital image processing. *Journal of information science and engineering*, pp. 164–173.
16. **Ghosh, S. & Chaudhury, K. N. (2016).** On fast bilateral filtering using fourier kernels. *IEEE Signal Processing Letters*, Vol. 23, pp. 570–573.
17. **Horn, B. P. K. (1986).** *Robot Vision*. New York, McGraw-Hill.
18. **Kuznetsova, A., Leal-Taixe, L., & Rosenhahn, B. (2013).** Real-time sign language recognition using a consumer depth camera. *Computer Vision Workshops (ICCVW), IEEE International Conference*, pp. 83–90.
19. **Lam, L., Lee, S.-W., & Suen, C. Y. (1992).** Thinning methodologies-a comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, pp. 869–885.
20. **Li, Y. (2012).** Hand gesture recognition using Kinect. *IEEE International Conference on Computer Science and Automation Engineering*, pp. 196–199.
21. **Liao, P., Chen, T., & Chung, P. (2001).** A fast algorithm for multilevel thresholding. *Journal of information science and engineering*, Vol. 17, pp. 713–727.
22. **Martin, J. & Crowley, J. L. (1997).** An appearance-based approach to gesture-recognition. *Proc. of 9th Conf on Image Analysis and Processing*, Italy.
23. **Ohn-Bar, E. & Manubhai Trivedi, M. (2014).** Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 15, pp. 2368–2377.
24. **Pandey, P. & Jain, V. (2015).** An efficient algorithm for sign language recognition. *International Journal of Computer Science and Information Technologies (IJCSIT)*, Vol. 6, pp. 5565–5571.
25. **Pratt, W. K. (1991).** *Digital Image Processing*. New York, John Wiley & Sons, Inc.
26. **Pugeault, N. & Bowden, R. (2011).** Spelling it out: Real-time asl fingerspelling recognition. *Proceedings of the 1st IEEE Workshop on Consumer Depth Cameras for Computer Vision*.
27. **Rautaray, S. & Agrawal, A. (2012).** Real time hand gesture recognition system for dynamic applications. *International Journal of UbiComp (IJU)*, Vol. 3.
28. **Ren, Z., Yuan, J., & Zhang, Z. (2011).** Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. *Proceedings of the 19th ACM international conference on Multimedia*, pp. 1093–1096.
29. **Soille, P. (1999).** *Morphological Image Analysis: Principles and Applications*. Springer-Verlag.
30. **Wang, C., Liu, Z., & Chan, S. (2015).** Superpixel-based hand gesture recognition with Kinect depth camera. *IEEE Transactions on Multimedia*, Vol. 17, pp. 29–39.

Ahmed BEN JMAA is received his M.Sc. degree in computer science from the National School of Engineers of Sfax, Sfax University, (Tunisia) in

2005. He is a Lecturer at the Department of computer sciences in Higher institute of business administration, Sfax University, (Tunisia). He is also a member of the Multimedia, InfoRmation systems and Advanced Computing Laboratory (MIRACL). His research interests include issues related to image processing of computer vision.

Walid MAHDI is received the Ph.D degree in Computer Science in 2001, from Ecole Centrale de Lyon (France) and the M.Sc. degree in Computer Science from Paris-Sorbonne University. He spent one year at Ecole Centrale-Lyon (France) as a PostDoc position. In 2004, he joined the Sfax University, Tunisia, Higher Institute of Computer Science and Multimedia, when he finished his habilitation, the highest academic qualification, in 2010. He is also a member of the Multimedia, InfoRmation systems and Advanced Computing Laboratory (MIRACL). His research interests include Computer Vision and Image and video analysis.

Yousra BEN JEMAA is received the engineering degree from Tunisia Polytechnic School (EPT) in 1997 and the M.Sc. degree in signal processing from Ecole Supérieure d'Electricité (SUPELEC), Paris, (France) in 1998. In 2003 She received the Ph.D. degree in Electrical Engineering and the HDR in Telecommunications in June 2012 from the National Engineers School of Tunis (ENIT), (Tunisia). She is a member of the research Lab. U2S at ENIT. Her teaching and research interests are in signal processing, image and video processing.

Abdelmajid BEN HAMADOU is received the Ph.D degree in computer science from the University of Orsay (France) in 1979 and a These d'Etat in Computer Science from the University of Tunis (Tunisia) in 1993. He is a Professor of Computer Science at the Higher Institute of Computer science and Multimedia, Sfax University, (Tunisia) and a member of the Multimedia, InfoRmation systems and Advanced Computing Laboratory (MIRACL). In 2002 he was decorated by the President of the Tunisian Republic ("Merit in Education and Science") and in 2009, he received from the Vice President of Syria the "Al-Kindi" Award ("the best computer science researcher"). His research interests include Natural Language Processing, emantic web, information retrieval/filtering and document summarizing.

Article received on 24/05/2016; accepted on 28/06/2016.
Corresponding author is Ahmed Ben Jmaa.