A Robust Kernel Descriptor for Finger Spelling Recognition Based on RGB-D Information

Karla Otiniano-Rodríguez #1, Guillermo Cámara-Chávez #2

# *Department of Computer Science (DECOM), Federal University of Ouro Preto*
Ouro Preto-MG-Brazil
1 karlaotiniano@gmail.com
2 gcamarac@gmail.com

***Abstract*—Systems of communication based on sign language and finger spelling are used by deaf people. Finger spelling is a system where each letter of the alphabet is represented by a unique and discrete movement of the hand. Intensity and depth images can be used to characterize hand shapes corresponding to letters of the alphabet. The advantage of depth sensors over color cameras for sign language recognition is that depth maps provide 3D information of the hand. In this paper, we propose a robust model for finger spelling recognition based on RGB-D information using a kernel descriptor. In the first stage, motivated by the performance of kernel based features, we decided to use the gradient kernel descriptor for feature extraction from depth and intensity images. Then, in the second stage, the Bag-of-Visual-Words approach is used to search semantic information. Finally, the features obtained are used as input of our Support Vector Machine (SVM) classifier. The performance of this approach is quantitatively and qualitatively evaluated on a dataset of real images of the American Sign Language (ASL) finger spelling. This dataset is composed of 120,000 images. Different experiments were performed using a combination of intensity and depth information. Our approach achieved a high recognition rate with a small number of training samples. With 10% of samples, we achieved an accuracy rate of 88.54% and with 50% of samples, we achieved a 96.77%; outperforming other state-of-the-art methods, proving its robustness.**

I. Introduction

Sign language is a complex way of communication in which hands, limbs, head and facial expression are used to communicate a visual-spatial language without sound, mostly used between deaf people. Deaf people use systems of communication based on sign language and finger spelling. In sign language, the basic units are composed by a finite set of hand configurations, spatial locations and movements. Their complex spatial grammars are remarkably different from the grammars of spoken languages [1], [2]. Hundreds of sign languages, such as ASL (American Sign Language), BSL (British Sign Language), Auslan (Australian Sign Language) and LIBRAS (Brazilian Sign Language) [1], are in use around the world and are at the cores of local deaf cultures. Unfortunately, these languages are barely known outside of the deaf community, meaning a communication barrier.

Finger spelling is a system where each letter of the alphabet is represented by a unique and discrete movement of the hand. Finger spelling integrates a sign language due to many reasons: when a concept lacks a specific sign, for proper nouns, for loan signs (signs borrowed from other languages) or when a sign is ambiguous [3]. Each sign language has its own finger spelling similar to different characters in different languages.

Several techniques have been developed to achieve an adequate recognition rate of sign language. Over the years and with the advance of technology, methods have been proposed in order to improve the data acquisition, processing or classification, such is the case of image acquisition. There are three main approaches: sensor-based, vision-based and hybrid systems using a combination of these systems. Sensor-based methods use sensory gloves and motion trackers to detect hand shapes and body movements. Vision-based methods, that use standard cameras, image processing, and feature extraction, are used for capturing and classifying hand shapes and body movements. Hybrid systems use information from vision-based camera and other type of sensors like infrared depth sensors.

Sensor-based methods, such as data gloves, can provide accurate measurements of hands and movement. Unfortunately, these methods require extensive calibration, they also restrict the natural movement of hands and are often very expensive. Video-based methods are less intrusive, but new problems arise: locating the hands and segmenting them is a non-trivial task. Recently, depth cameras have become popular at a commodity price. Depth information makes the task of segmenting the hand from the background much easier. Depth information can be used to improve the segmentation process, as used in [4], [5], [6], [7].

Recently, depth cameras have raised a great interest in the computer vision community due to their success in many applications, such as pose estimation [8], [9], tracking [10], object recognition [10], etc. Depth cameras were also used for hand gesture recognition [11], [12], [13], [14], [15]. Uebersax et al. [12] present a system for recognizing letter and finger spelled words. Pugeault & Bowden [11] use a Microsoft Kinect^{TM} device to collect RGB and depth images. They extracted features using Gabor filters and then a Random Forest predicts the letters from the American Sign Language

(ASL) finger spelling alphabet. Issacs & Foo [16] proposed an ASL finger spelling recognition system based on neural networks applied to wavelets features. Bergh & Van Gool [17] propose a method based on a concatenation of depth and color-segmented images, using a combination of Haar wavelets and neural networks for 6 hand poses recognition of a single user.

In this paper, we propose a framework for finger spelling recognition using intensity and depth images. Motivated by the performance of kernel based features, due to its simplicity and the ability to turn any type of pixel attribute into patch-level features, we decided to use the gradient kernel descriptor [18]. The experiments are performed using a public database composed of 120,000 images stating 24 symbols classes [19]. The obtained results show that the accuracy obtained by our method, using intensity and depth images, is greater than only using intensity or depth images separately. Moreover, the accuracy obtained by the proposed method performs better than the methods proposed in [11], [15]. The results show that our method is promising.

The remainder of this paper is organized as follows. In Section II, our proposed method is introduced and detailed. The experiments are presented in Section III, where the results are discussed. Finally, conclusion and future work are presented in Section IV.

II. Proposed Model

This section describes the methodology developed to perform a finger spelling recognition from RGB-D information. The proposed model consists of two stages as shown in Figure 1. In the first stage, we apply the bag-of-visual-words approach, this technique consists of three steps, feature description, vocabulary generation and histogram generation. For feature extraction, we use intensity and depth images and the gradient kernel descriptor is applied on those images. This kernel descriptor consists of three kernels. The normalized linear kernel weighs the contribution of each pixel using gradient magnitudes, an orientation kernel computes the similarity of gradient orientations and finally a position Gaussian kernel measures how close two pixels are spatially. The grouping by similarity of features extracted in the previous step generates the visual vocabulary, the centroid of each group represents a visual word. Thus, the visual words histogram is obtained by counting the number of occurrences of each visual word. Finally, in the second stage, these histograms are used as input to our SVM classifier.

A. *Bag-of-Visual-Words*

Bag-of-Visual-Words has first been introduced by Sivic for video retrieval [20]. Due to its efficiency and effectiveness, it became very popular in the fields of image retrieval and categorization. Image categorization techniques rely either on unsupervised or supervised learning.

Our model uses the Bag-of-Visual-Words approach in order to search semantic information. The original method works with documents and words. Therefore, we consider an image as a document and the ”words” will be the visual entities found in the image. The Bag-of-Visual-Words approach consists of three operations: feature description, visual word vocabulary generation and histogram generation.

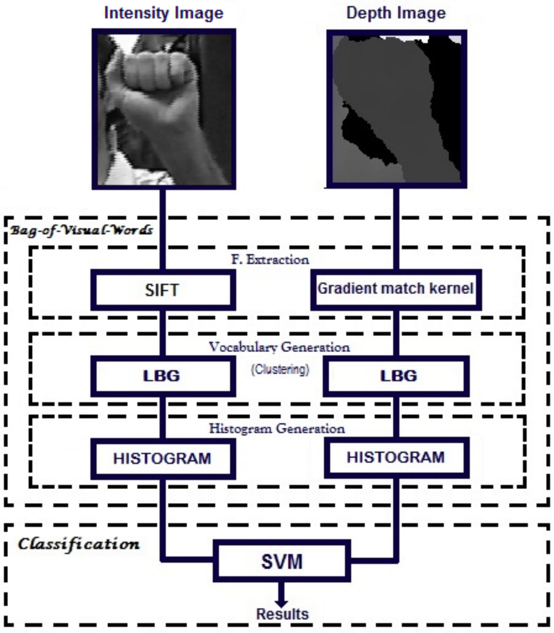

Fig. 1. Proposed model for finger spelling recognition.

1) Feature Description: Gradient Kernel Descriptor: The low-level image feature extractor, kernel descriptor, designed for visual recognition in [21], consists of three steps: design match kernel using some pixel attribute, learn compact basis vectors using Kernel Principle Component Analysis and construct kernel (KPCA) descriptor by projecting the infinite-dimensional feature vector to the learned basis vectors. The authors proposed three types of effective kernel descriptors using gradient, color and shape pixel attributes. In other model proposed by the same authors [18], the gradient kernel descriptor is applied over depth images. Thereby, in order to capture edge cues in depth maps, we used the gradient match kernel, K_{grad} :

$$K_{grad}(P,Q)=\sum_{p\in P}\sum_{q\in Q}\tilde{m}(p)\tilde{m}(q)k_o(\tilde{\theta}(p),\tilde{\theta}(q))k_s(p,q) \tag{1}$$

The normalized linear kernel $\tilde{m}(p)\tilde{m}(q)$ weighs the contribution of each gradient where $\tilde{m}(p) = m(p)/\sqrt{\sum_{p\in P} m(p)^2+\varepsilon_g}$ and ε_g is a small positive constant to ensure that the denominator is larger than 0 and $m(p)$ is the magnitude of the depth gradient at a pixel p. Then, $k_o(\tilde{\theta}(p),\tilde{\theta}(q)) = exp(-\gamma_o\|\tilde{\theta}(p)-\tilde{\theta}(q)\|^2)$ is a Gaussian kernel over orientations. The authors [21] suggest to set $\gamma_o = 5$. To estimate the difference between orientations at pixels p and q, we use the following normalized gradient vectors in the kernel function k_o:

$$\tilde{\theta}(p) = [sin(\theta(p))cos(\theta(p))]$$
$$\tilde{\theta}(q) = [sin(\theta(q))cos(\theta(q))]$$

where $\theta(p)$ is the orientation of the depth gradient at a pixel p. Gaussian position kernel $k_s(p,q) = exp(-\gamma_s\|p - q\|^2)$ with p denoting the 2D position of a pixel in an image patch (normalized to [0,1]), measures how close two pixels are spatially. The value suggest for γ_s is 3.

To summarize, the gradient match kernel K_{grad} consists of three kernels: the normalized linear kernel weighs the contribution of each pixel using gradient magnitudes; the orientation kernel k_o computes the similarity of gradient orientations; and the position Gaussian kernel k_s measures how close two pixels are spatially.

Match kernels provide a principled way to measure the similarity of image patches, but evaluating kernels can be computationally expensive when image patch are large [21]. The corresponding kernel descriptor can be extracted from this match kernel by projecting the infinite-dimensional feature vector to a set of finite basis vectors, which are the edge features that we use in the next steps. For more details, the approach that extracts the compact low-dimensional features from match kernels is found in [21].

2) Vocabulary Generation: Then, a visual word vocabulary is generated from the feature vectors;s each visual word (codeword) represents a group of several similar features. The visual word vocabulary (codebook) defines a space of all entities occurring in the image.

3) Histogram Generation: Finally, a histogram of visual words is created by counting the occurrence of each codeword. These occurrences are counted and arranged in a vector. Each vector represents the features for an image.

B. Classification

Support vector machines, introduced as a machine learning method by Cortes and Vapnik [22], are a useful classification method. Furthermore, SVMs have been successfully applied in many real world problems and in several areas: text categorization, handwritten digit recognition, object recognition, etc. The SVMs have been developed as a robust tool for classification and regression in noisy and complex domains. SVM can be used to extract valuable information from data sets and construct fast classification algorithms for massive data.

An important characteristic of the SVM classifier is to allow a non-linear classification without requiring explicitly a non-linear algorithm thanks to kernel theory.

In kernel framework data points may be mapped into a higher dimensional feature space, where a separating hyperplane can be found. We can avoid to explicitly computing the mapping using the kernel trick which evaluate similarities between data $K(d_t, d_s)$ in the input space. Common kernel functions are: linear, polynomial, Radial Basis Function (RBF), χ^2 distance and triangular.

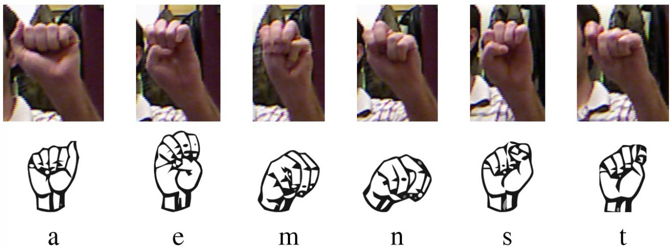

Fig. 3. Most conflictive similar signs in the dataset.

III. Experiments

The ASL Finger Spelling Dataset [19] contains 500 samples for each of 24 signs, recorded from 5 different persons (non-native to sign language), amounting to a total of 60,000 samples. Each sample has a RGB image and a depth image, making a total of 120,000 images. The sign J and Z are not used, because these signs have motion and the proposed model only works with static signs. The dataset has variety of background and viewing angles. Figure 2 shows some examples and there is possible to see the variety in size, background and orientation.

Due to the variety in the orientation when the signal is performed, signs became strongly similar. Figure 3 shows the most similar signs *a*, *e*, *m*, *n*, *s* and *t*. The examples are taken from the same user. It is easy to identify the similarity between these signs, all are represented by a closed fist, and differ only by the thumb position, leading to higher confusion levels. Therefore, these signs are the most difficult to differentiate in the classification task.

In order to validate our technique, we conduct three experiments. In the first, a classification of the signs was performed using different percentages of samples for training and testing from intensity information. In the second, a classification was also performed from depth information varying the percentages of training and testing. Finally, a classification of the signs was performed using different percentages of samples for training and testing from both information (RGB-D).

For each experiment, we have some specifications:

- To extract all low level features using gradient kernel descriptor, are used approximately 12x13 patches over dense regular grid with spacing of 8 pixel (images are not of uniform size), each patch has a size of 16x16.
- In order to produce the visual word vocabulary, the LBG (Linde-Buzo-Gray) [23] algorithm was used to detect one hundred clusters by taking a sample of 30% from the total features.
- Moreover, in the classification stage, we use a RBF kernel, whose values for *g* (gamma) and *c* (cost) are 0.25 and 5, respectively. We also use different percentages of samples for training and testing. For example, we use 10% of samples for training and the other 90% is used to testing, and this percentage varies up to 50% for training. In order to obtain more precise results, each experiment

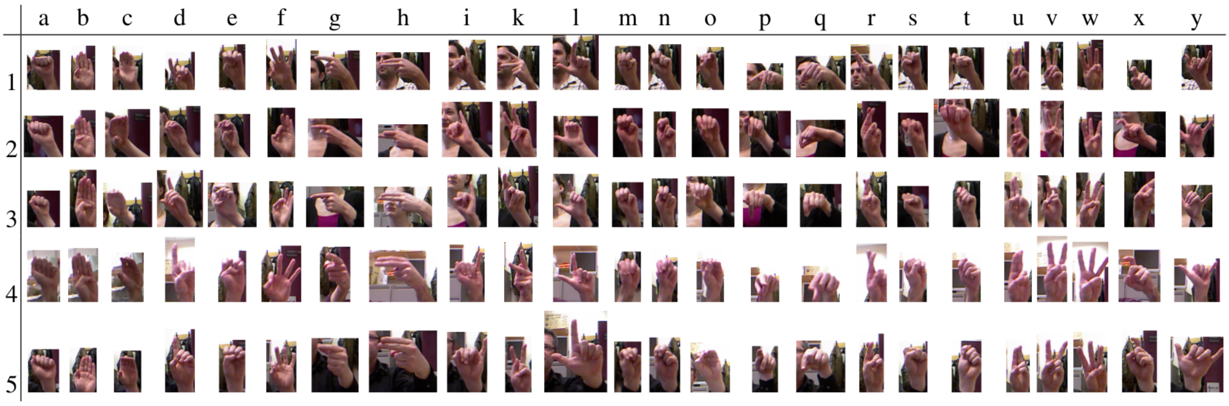

Fig. 2. ASL Finger Spelling Dataset: 24 static signs by 5 users. It is an example of the variety of the dataset. This array shows one image from each user and from each letter.

TABLE I
ACCURACIES AND STANDARD DEVIATION OF THE CLASSIFICATION USING INTENSITY INFORMATION.

% Training	% Testing	Accuracy	Standard deviation
10	90	79.08%	0.25
20	80	85.28%	0.21
30	70	88.32%	0.20
40	60	90.19%	0.15
50	50	91.58%	0.16

TABLE II
ACCURACIES AND STANDARD DEVIATION OF THE CLASSIFICATION USING DEPTH INFORMATION

% Training	% Testing	Accuracy	Standard deviation
10	90	75.60%	0.26
20	80	81.18%	0.21
30	70	84.24%	0.17
40	60	85.54%	0.19
50	50	86.86%	0.17

was performed 30 times and we show the mean accuracy for each one. The library LIBSVM (a library for Support Vector Machines)] [24] was used in our implementation.

First experiment: An average accuracy of 79.08% when a 10% of samples are used for training and 90% for testing. This accuracy is the mean of the values of the main diagonal of the confusion matrix and represents the signs correctly classified (true positives). This accuracy increases when more samples are used for training. With 30% of samples for training is obtained 88.32% and when 50% of samples are training we obtain 91.58% of accuracy. More results are found in the Table I. The classification using intensity information was improved compared to the proposed model found in [15], in which, was obtained an accuracy of 62.70% using the same type of information.

Second experiment: For this experiment, using depth information, the average accuracy obtained was 75.6% when 10% of samples are used for training. The higher accuracy, 86.86%, was obtained using 50% of samples for training and the other 50% of samples for testing. Other results are found in the Table II. This results show a slight increase in the classification rate compared to the results found in [15], where was obtained an accuracy of 85.18%.

Third experiment: The classification task was performed using RGB-D information. The data for this experiment was obtained by joining the features (histograms) from RGB and depth information, which were used in the experiments 1 and 2, respectively. Is obtained an average accuracy of 96.77% when 50% of samples are used for training. In other case, when are used 10% of samples for training, we obtain an accuracy of 88.54%. It means that are used 250 samples for each sign for training and 2250 samples for testing. Table III shows the results for this case (10% to training). Signs *f*, *b*, *l* and *y* have the highest average accuracies (over 95%). Otherwise, the signs *n*, *m*, *r* and *t* have the lowest values (with 80% and 81%). The low recognition value of sign *t* is due to the big similarity with signs *m* and *n*, as shown in the Figure 3. Table IV shows the results when 50% of samples are used for training. In the similar case, the signs with highest accuracies *a*, *b*, *f* and *l* have 99% of recognition. Otherwise, signs *t*, *v*, *m* and *n* have an accuracy between 93% and 94%. However, each sign have an accuracy greater than 93%, proving the high recognition rate of our proposed model. In Table V are found the average accuracies for each experiment using different percentages of samples for training.

We summarize and compare the results in Tables VI and VII. It includes the average accuracy and standard deviation for each experiment. We can see that using RGB-D information we obtain the highest average accuracy, outperforming the intensity and depth methods and also the methods proposed by Pugeault & Bowden [11] and Zhu & Wong [14]. These last methods are found in the state-of-the-art and use the same dataset, the principal difference between these methods is the number of samples used for training. Pugeault & Bowden [11]

TABLE III
CONFUSION MATRIX OF THE CLASSIFICATION OF 24 SIGN USING RGB-D INFORMATION WITH 10% FOR TRAINING AND 90% FOR TESTING.

	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
a	0.94	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00
b	0.01	0.95	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00
c	0.01	0.01	0.93	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
d	0.00	0.00	0.00	0.88	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.02	0.00	0.00	0.01	0.01	0.00	0.01	0.00
e	0.03	0.00	0.01	0.01	0.85	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.02	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.01	0.00
f	0.00	0.00	0.00	0.00	0.00	0.97	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
g	0.01	0.00	0.00	0.00	0.00	0.00	0.91	0.05	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
h	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.94	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
i	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.92	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01
k	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.88	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.01	0.02	0.02	0.00	0.01	0.00
l	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
m	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.81	0.09	0.00	0.00	0.00	0.01	0.03	0.01	0.00	0.00	0.00	0.00	0.00
n	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.06	0.81	0.01	0.00	0.00	0.00	0.02	0.03	0.00	0.00	0.00	0.02	0.00
o	0.00	0.00	0.01	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.01	0.85	0.01	0.01	0.00	0.02	0.01	0.00	0.00	0.00	0.01	0.00
p	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.92	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
q	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.92	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
e	0.01	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.80	0.00	0.00	0.04	0.06	0.00	0.02	0.00
s	0.01	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.00	0.03	0.05	0.01	0.00	0.00	0.00	0.83	0.01	0.00	0.00	0.00	0.01	0.00
t	0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.03	0.08	0.01	0.00	0.00	0.00	0.02	0.80	0.00	0.00	0.00	0.01	0.00
u	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.05	0.00	0.00	0.85	0.04	0.02	0.00	0.00
v	0.01	0.01	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.04	0.00	0.00	0.05	0.82	0.03	0.00	0.00
w	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.03	0.93	0.00	0.00
x	0.00	0.00	0.01	0.02	0.01	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.02	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.86	0.00
y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.95

TABLE IV
CONFUSION MATRIX OF THE CLASSIFICATION OF 24 SIGN USING RGB-D INFORMATION WITH 50% OF SAMPLES FOR TRAINING AND 50% FOR TESTING.

	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
a	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
b	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
c	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
d	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
e	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
f	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
g	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
h	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
i	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
k	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00
l	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
m	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.94	0.03	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
n	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.94	0.01	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00
o	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.97	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
p	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
q	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
r	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.94	0.00	0.00	0.02	0.01	0.00	0.00	0.00
s	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.96	0.01	0.00	0.00	0.00	0.00	0.00
t	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.01	0.93	0.00	0.00	0.00	0.00	0.00
u	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.94	0.01	0.00	0.00	0.00
v	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.03	0.93	0.01	0.00	0.00
w	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.97	0.00	0.00
x	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97	0.00
y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98

TABLE V
ACCURACIES AND STANDARD DEVIATION OF THE CLASSIFICATION USING RGB-D INFORMATION

% Training	% Testing	Accuracy	Standard deviation
10	90	88.54%	0.17
20	80	93.17%	0.14
30	70	95.02%	0.11
40	60	96.22%	0.12
50	50	96.77%	0.09

TABLE VI
ACCURACIES AND STANDARD DEVIATION OF THE THREE EXPERIMENTS USING 10% OF SAMPLES FOR TRAINING.

Method	Accuracy	Standard Deviation
RGB	79.08%	0.25
Depth	75.6%	0.26
RGB-D	88.54%	0.17
Zhu & Wong [14]	88.9%	0.39

use the 50% of samples (1250 samples) for training and Zhu & Wong [14] use only 40 samples for training. In the Table VI are found results when 10% of samples are used to training with each type of information and the result for the method proposed by Zhu & Wong [14]. In the Table VII are found the accuracies for each experiment when 50% of samples are used for training, also this is the case of the method proposed by Pugeault & Bowden [11].

IV. CONCLUSION AND FUTURE WORK

In this paper, we propose a method for Finger Spelling Recognition from RGB-D information using a robust kernel descriptor. Then, Bag-of-Visual-Words was applied in order to search semantic information. Finally, the classification task is performed by a SVM. The combination of RGB and depth descriptors obtains the best results (96.77%) with a low variance. Our method achieves a better differentiation of similar signs like *n*, *r* and *t*, incrementing the recognition rate. The Gradient kernel descriptor has the advantage that can be directly applied on the depth images without having to compute the cloud of points, consequently, reducing the computation time. In a previously proposed model [15], we used segmentation to better detect the hand. Even though in this paper we do not segment the images, we obtain better results, showing the robustness of kernel descriptors. As future work, we pretend to test other kernels over depth and intensity images. We also intend to extend our method to recognize dynamic signs.

TABLE VII
ACCURACIES AND STANDARD DEVIATION OF THE THREE EXPERIMENTS USING 50% OF SAMPLES FOR TRAINING.

Method	Accuracy	Standard Deviation
RGB	91.58%	0.16
Depth	86.85%	0.17
RGB-D	96.77%	0.09
Pugeault & Bowden[11]	75.00%	-

V. ACKNOWLEDGMENT

The authors are thankful to CNPq, CAPES and FAPEMIG (Projeto Universal 02292-12), Brazilian funding agencies and to the Federal University of Ouro Preto (UFOP) for supporting this work.

REFERENCES

[1] LIBRAS, "Brazilian sign language," http://www.libras.org.br/, last visit: March 10, 2012.

[2] P. W. Vamplew, "Recognition of sign language gestures using neural networks," *Australian Journal of Intelligent Information Processing Systems*, vol. 5, pp. 27–33, 1996.

[3] A. Puente, J. M. Alvarado, and V. Herrera, "Fingerspelling and sign language as alternative codes for reading and writing words for Chilean deaf signers," *American Annals of the Deaf*, vol. 151, no. 3, pp. 299–310, 2006.

[4] Z. Ren, J. Yuan, and Z. Zhang, "Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera," in *Proceedings of the 19th ACM International Conference on Multimedia*. ACM, 2011, pp. 1093–1096.

[5] V. Frati and D. Prattichizzo, "Using Kinect for hand tracking and rendering in wearable haptics," in *Proceedings of the IEEE World Haptics Conference (WHC)*. IEEE, 2011, pp. 317–321.

[6] Y. Li, "Hand gesture recognition using Kinect," in *Proceedings of the 3rd IEEE International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2012, pp. 196–199.

[7] Z. Mo and U. Neumann, "Real-time hand pose recognition using low-resolution depth images," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2006, pp. 1499–1505.

[8] G. Fanelli, J. Gall, and L. V. Gool, "Real time head pose estimation with random regression forests," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 617–624.

[9] J. Shotton, T. Sharp, A. Kipman, A. W. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, "Real-time human pose recognition in parts from single depth images," *Communications of the ACM*, vol. 56, no. 1, pp. 116–124, 2013.

[10] I. Oikonomidis, N. Kyriazis, and A. Argyros, "Efficient model-based 3D tracking of hand articulations using Kinect," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2011, pp. 101.1–101.11.

[11] N. Pugeault and R. Bowden, "Spelling it out: Real-time ASL fingerspelling recognition." in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. IEEE, 2011, pp. 1114–1119.

[12] D. Uebersax, J. Gall, M. V. den Bergh, and L. J. V. Gool, "Real-time sign language letter and word recognition from depth data," in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 383–390.

[13] M. d. S. Anjo, E. B. Pizzolato, and S. Feuerstack, "A real-time system to recognize static gestures of brazilian sign language (libras) alphabet using kinect," in *Proceedings of the 11th Brazilian Symposium on Human Factors in Computing Systems*. Brazilian Computer Society, 2012, pp. 259–268.

[14] X. Zhu and K.-Y. K. Wong, "Single-frame hand gesture recognition using color and depth kernel descriptors," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*. IEEE, 2012, pp. 2989–2992.

[15] K. Otiniano-Rodríguez and G. Cámara-Chávez, "Finger spelling recognition from RGB-D information using kernel descriptor," in *Proceedings of the SIBGRAPI 2013 (XXVI Conference on Graphics, Patterns and Images)*, 2013.

[16] J. Isaacs and S. Foo, "Hand pose estimation for american sign language recognition," *36th Southeastern Symposium on System Theory*, pp. 132–136, 2004.

[17] M. Van den Bergh and L. Van Gool, “Combining RGB and ToF cameras for real-time 3D hand gesture interaction,” in *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV)*, ser. WACV ’11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 66–72.

[18] L. Bo, X. Ren, and D. Fox, “Depth kernel descriptors for object recognition,” in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2011, pp. 821–826.

[19] R. B. Nicolas Pugeault, “ASL finger spelling dataset,” http://personal.ee.surrey.ac.uk/Personal/N.Pugeault/index.php, last visit: April 29, 2013.

[20] J. Sivic and A. Zisserman, “Video google: A text retrieval approach to object matching in videos,” in *Proceedings of the Ninth IEEE International Conference on Computer Vision*. IEEE, 2003, pp. 1470–1477.

[21] L. Bo, X. Ren, and D. Fox, “Kernel descriptors for visual recognition,” *Advances in Neural Information Processing Systems*, vol. 7, 2010.

[22] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[23] Y. Linde, A. Buzo, and R. Gray, “An algorithm for vector quantizer design,” *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84–95, 1980.

[24] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.