# Static hand gesture recognition

Report*

Thierry Messer†
Department of Informatics
University of Fribourg
1700 Fribourg
Switzerland
thierry.messer@unifr.ch

## ABSTRACT

This survey presents an overview of the challenging field of static hand gesture recognition, which mainly consists of the recognition of well defined signs based on a posture of the hand. Since human beings tend to differ in terms of size and shape, the most challenging problem consists of the segmentation and the correct classification of the informations gathered from the input data, captured by one or more cameras. The aim of this report is to show which techniques have successfully been tested and used in order to solve the problems mentioned above yielding a robust and reliable static hand gesture recognition system.

## Keywords

static gesture recognition, hand posture

## 1. INTRODUCTION

Hand gestures recognition provides a natural way to interact and communicate with machines of different kinds. Compared to the currently used human-machine-interfaces (HMI) such as a keyboard or a remote control, static hand gesture recognition does without any supplementary devices which are used to give instructions to a machine. In a process, which is generally known and referred to as *static hand gesture recognition*, a person instructs the machine using his bare hands, whereas images of the persons hand gestures are captured and analyzed in order to determine the meaning of the hand gesture.

Although a static hand gesture could theoretically be any possible posture of a humans hand, usually only a limited set of well defined postures are considered to be used in the communication since similarities between postures with different meaning tend to raise the number of wrong detected / interpreted gestures, and thus the error rate.
In general, gesture recognition is considered as a very challenging field since natural environments tend to be rather unsuitable for gesture recognition due to bad illumination, nonuniform backgrounds, and so on. The numerous publications of the recent years show that static hand gesture recognition is still a field of active research, whereas many of them try to face the previously mentioned problems in order to improve the performance and quality of existing technologies.
There exist several additional devices (e.g. data gloves), which are used to solve the previously mentioned problems by providing a more precise capturing of the hand information. However, this report refers only to camera based static hand gesture recognition.

A possible application of static hand gesture recognition is the machine aided communication using the American Sign Language (ASL) in order to allow the communication between ASL- and non-ASL-speakers.
Yet another application involves the control of consumer electronics, such as TVs, HiFi-systems, DVD/CD players and so on. A user could therefore use some control gestures in order to switch them off or on, change the radio or TV program or to select some movie or music. Combinations of gestures could even be used to perform more complex tasks, such as scheduling the recording of one's favorite TV-show.

This report aims to give an overview of the technologies and methods used to recognize static hand posture recognition. The next section summarizes the basic principles of static hand gesture recognition and shows the technologies that are used for all the different tasks, also discussing the advantages and disadvantages of each technology. In section 3 some applications are presented, while section 4 covers the discussion. The last section contains the conslusions.

## 2. GESTURE RECOGNITION PROCESS

There are two basic approaches in static gesture recognition, as described in [8]:

1. The top-down approach, where a previously created

*This report was created in the context of a master seminar about gesture recognition. For more information please visit the official seminar website: `http://diuf.unifr.ch/diva/web/site/index.php/teaching-seminars/10-seminars/125-gesture-recognition`

†MSc. Student

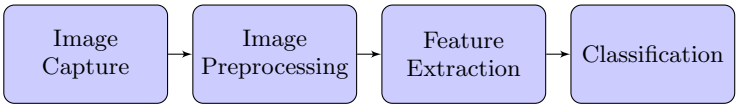


**Figure 1: Schematic view of gesture recognition process**

model of collected informations about hand configurations is rendered to some feature in the image coordinates. Comparing the likelihood of the rendered image with the real gesture image is then used to decide whether the gesture of the real image corresponds to the rendered one.

2. The bottom-up approach, which extracts features from an input image and uses them to query images from a database, where the result is based on a similarity measurement of the database image features and the input features.

The disadvantage of the first approach is that it seems to use a high computational effort in order to achieve robust recognition. The second approach however requires an adequate preprocessing in order to achieve a reliable segmentation. This report mainly keeps the focus on the latter approach since this seems to be the commonly used one.

The whole process of static gesture recognition can be coarsely divided into four phases, as shown in Figure 1. Each phase performs a specific task, whose result is passed to the next phase. The commonly used techniques for each phase are descibed in the following subsections.

## 2.1 Image capturing

The task of this phase is to aquire an image, or a sequence of images (video), which is then processed in the next phases. The capturing is mostly done using a single camera with a frontal view of the persons hand, which performs the gestures. However, there also exist systems that use two or more cameras in order to aquire more informations about the hand posture [5, 8]. The advantage of such a system is that it allows a recognition of the gesture, even if the hand is occluded for example by the body of the person that performs the gesture, since the other camera captures the scene from another perspective.
Yet another system was presented in [6], where the camera was mounted on a hat, capturing the area in front of the wearer. Clearly the advantage of this system is that the camera position is always adapted if the person moves or turns his body around.

In general, the following phases of the recognition process are less complex if the captured images do not have cluttered backgrounds, although several recognition systems [6, 1] seem to work reliable even on cluttered images. Therefore, the image capturing is often performed in a cleaned up environment having a uniform background [3]. It is also desirable to have an equalized distribution of luminousity in order to gather images without shadowy regions.

## 2.2 Preprocessing

The basic aim of this phase is to optimally prepare the image obtained from the previous phase in order to extract the features in the next phase. How an optimal result looks like depends mainly on the next step, since some approaches only need an approximate bounding box of the hand, whereas others need a properly segmented hand region in order to get the hand silhouette. In general, some regions of interest, that will be subject of further analysis in the next phase, are searched in this phase.

The most commonly used technic to determine the regions of interest is skin color detection [7, 2, 8]. A previously created probabilistic model of skin-color is used to calculate the probability of each pixel to represent some skin. Thresholding then leads to the coarse regions of interest. Some further analysis could for example involve the size or perimeter of the located regions in order to exclude regions such as the face.

Other systems, as described [6], initially search the image for a pixel of a specific color using the eight nearest neighbors of the appropriate color in order to start the growth of the region. For subsequent images the center of the regions detected in the previous image is used to find the hand regions.

Yet another interesting approach is to use a previously acquired image of the background, substracting it from the image with the gesture, as proposed in [5]. Based on perimeter lengths, the hand region can then be extracted.

## 2.3 Feature extraction

The aim of this phase is to find and extract features that can be used to determine the meaning of a given gesture. Some interesting techniques are presented later on in this section. Ideally such a feature, or a set of such features, should uniquely describe the gesture in order to achieve a reliable recognition. Therefore, different gestures should result in different, good discriminable features. Furthermore, shift and rotation invariant fetures lead to a better recognition of hand gestures even if the handgesture is captured in a different angle.

*Hand outline.* This is a simple approach which relies on the outline of a given hand region [5]. Given a hand region the outline is extracted using for example some edge tracking algorithm. The local features are then represented by the local extrema of the outline, whereas there are two different kind of extrema: The peaks and the valleys. The peaks are usually found at the finger tips, whereas the valleys are rather found in the regions where two fingers join the palm of the hand. One advantage of such features is the quick exclusion of inappropriate gestures, using the number

of peaks and valleys as indicators. A disadvantage of this approach is the relatively small number of different gestures that can be distinguished, since only considering the outline does not permit using the fingers' actual position. Therefore, it is for example not possible to distinguish between two hand postures, where one uses the middle and the ring finger, while the other hand uses the ring and the fore finger. As a result, this method only works well in an environment where only few gestures have to be distinguished, as it is the case in [5].

*Zernike moments.* Zernike moments (ZM) and pseudo Zernike moments (PZM) are in general used to describe shapes, whereas ZMs are usually better for describing shapes than PZMs. On the other hand, PZMs are known to be less affected by noise. In order to use ZMs for hand features description, the hand is represented as a set of ZMs rather than using a single ZM. In [3], they proposed to first separate the hand into two sub-regions, where one region contains the finger part, and the other consists of the palm. The ZMs and PZMs are then calculated for each finger and for the palm, using the center of the minimum bounding circle of the hand silhouette, which has the advantage of translation invariance, making this feature more reliable. Another important technique, that is presented in [3], uses a different weight for the palm and the finger features. Since most gestures depend more on the actual positions of the fingers and less on the palm position, the weight for the fingers should be bigger than weight for the palm region. Empirical tests lead to a weight of 0.7 for the finger features and 0.3 for the palm feature, for which the best results were obtained.

*Local Orientation Histogram.* In [8], the utilization of so called Local Orientation Histogram features is proposed. In general, orientation histograms cannot be directly applied to hand gestures as the hand does not provide sufficient texture. Since orientation histograms show the frequency of edges aligned in a certain angle, there might be not enough information available inside the hand area in order to uniquely describe a hand gesture. According to [4], the main problem that might arise is that hand gestures which look different for a human being, might have almost identical orientation histograms. Yet another problem is that hand gestures which look very similar for humans (for example a rotation of the hand) can yield very different orientation histograms. However, in [8] it is found that the boundary of the hand shape contains enough information to uniquely describe the feature of a specific gesture. Therefore, the idea of local orientation histograms consists of creating overlapping subwindows, whereas each subwindow contains at least one pixel which lies inside the hand shape. For each of these subwindows an orientation histogram is created, which is then added to the feature vector. Beside the local orientation histograms also the subwindow positions are added to the feature vector. These positions are measured relative to the median value of all pixel positions that were determined to be in the hand region. Clearly, the advantage of this technique lies in the improved robustness since using relative positions allow in-plane translations.

*Multi scale color features.* Multi scale color features, as used in [2], do not require any preprocessing of the image. Multi scale features can be found in an image at different scales. Therefore, the hand can be described as one bigger blob feature for the palm, having smaller blob features representing the finger tips which are connected by some rigid features. Thus, the hand can be detected in the image without having properly segmented the hand region since blob- and rigid-feature occurrences are found in different sizes. Furthermore, it was proposed to perform the feature extraction directly in the color space, as this allows the combination of probabilistic skin-colors directly in the extraction phase. The advantage of directly working on a color image lies in the better distinction of hand and background regions.

## 2.4 Classification

The classification represents the task of assigning a feature vector or a set of features to some predefined classes in order to recognize the hand gesture. In previous years several classification methods have been proposed and sucessfully tested in different recognition systems. In general, a class is defined as a set of reference features that were obtained during the training phase of the system or by manual feature extraction, using a set of training images. Therefore, the classification mainly consists of finding the best matching reference features for the features extracted in the previous phase. This section presents an overview of the most commonly used methods in different hand gesture recognition systems.

*k-Nearest Neighbors.* This classification method uses the feature-vectors gathered in the training to find the k nearest neighbors in a n-dimensional space. The training mainly consists of the extraction of (possible good discriminable) features from training images, which are then stored for later classification. Due to the use of distance measuring such as the euclidian or manhattan distance, the algorithm performs relatively slowly in higher dimensional spaces or if there are many reference features. In [8], an approximate nearest neighbors classification was proposed, which provides a better performance.

*Hidden Markov Models.* The Hidden Markov Model (HMM) classifiers belong to the class of trainable classifiers. It represents a statistical model, in which the most probable matching gesture-class is determined for a given feature vector, based on the training data. In [6], HMMs were successfully used to distinguish up to 40 different hand gestures with an accuracy of up to 91.9%.
In order to train the HMM, a Baum-Welch re-estimation algorithm, which adapts the internal states of the HMM according to some feedback concerning the accuracy, was used.

*Multi Layer Perceptron.* A Multi Layer Perceptron (MLP) classifier is based on a neural network. Therefore, MLPs represent a trainable classifier (similar to Hidden Markov Models). They use three or more layers of neurons that are all connected. During the training phase, the weights of the connections between the neurons are adapted, based on the

feedback that describes the difference between the output and the expected result. In [7], a MLP classifier was used to recognize 26 different ASL gestures with a recognition rate of up to 98.7%, depending on the number of features used to describe the guesture.

# 3. APPLICATIONS

In this section two example systems that show different possible static gesture recognition applications are presented.

## 3.1 ASL recognition

In [6], two static gesture recognition systems that are used in order to recognize the ASL gestures are described. The first one is a desk based system, where the signing person is captered using a frontal view. The second system is wearable, whereas the camera is mounted on a hat as described in section 2.1. For both systems the same recognition system was used, which is based on a HMM classification. In order to train and test both systems a 500 sentences database, of which 400 were used for training and 100 for testing, was used. The 500 senctences were constructed out of a 40 gestures grammar. For the second system an additional gesture "silence" was introduced, which describes situations where the hands are in rest or if no hand could be detected in the image. The new gesture became necessary since turning the head while performing gestures can lead to such images.
The tests showed a recognition rate of about 97% for the wearable system and around 92% for the desk based system. A possible explication for the better rate of the wearable system is that there is less occlusion between both hands or the face. Another explanation is that the wearable system automatically compensates body rotation. However, the authors mentioned that 25% of all errors were insertion errors, caused by repeated recognition of the same gesture.

## 3.2 3D gesture recognition system

The aim of the system suggested in [5] is to constitute a more natural kind of input device, which can be used to navigate in 3D environments. This gesture recognition system represents a possible application of a rather simple functionality which only distinguishes 4 gestures: Point, Reach, Click and Ground, whereas only Point and Reach are static gestures. However, the Click gesture can also be regarded as a static gesture since its recognition depends only on a preceeding Point gesture. In contrast to the other three gestures the Ground gesture is a pseudo gesture which is used for any unrecognized hand posture or in case the image did not include a hand at all. The system is based on two desk mounted cameras that capture the frontal view of the user, each from a different angle. Using both images of the same gesture allows the system to extract a direction vector, representing the pointing direction.
In order to demonstrate the various possibilities the system actually provides, it was used to create composite objects in a 3D editor environment. Furthermore, a virtual reality fly-through over a 3D terrain was presented, using gesture based navigation. The last mentioned example applied the system in the famous 3D video game Doom[1]. As an overall statement the system was considered to be very stable and test-users claimed that the system is more intuitive and easy to use compared to a mouse or a keyboard.

[1]Doom, a 3D video game by id Software, `http://www.idsoftware.com`

# 4. DISCUSSION

In [6], there seem to be some problems concerning multiple recognition of a single gesture when using the nonrestricted grammar. While missing a real explanation for the phenomenon it is indicated that this happens when a gesture is performed over a relatively long period of time. However, it remains unclear how a changeover between two hand gestures is detected and whether this could be a solution to the previously mentioned problem.
Also in the other articles, either it is not explicitly mention how such a changeover is detected or a changeover is not mentioned at all. An only exlanation is proposed in [2], where particle filtering is used to track the hand position and a posture change is defined as a random variation of more than 30% of all particles.

# 5. CONCLUSIONS

In the past, several real-time gesture recognition systems, for example the ASL recognizer in [6], have been presented, which turned out to operate accurate on a relatively small set of gestures.
The main problem of static gesture recognition lies in the complexity of the classification algorithms, especially when using high dimensional feature vectors which become necessary in order to be able to distinguish several hundreds of gestures. Thus, the development of faster classification methods and more accurate and precise features is very important in order to run such systems in real-time.

# 6. REFERENCES


[1] V. Athitsos and S. Sclaroff. Estimating 3d hand pose from a cluttered image. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 432–439, June 2003.

[2] L. Bretzner, I. Laptev, and T. Lindeberg. Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering. In *Proc. Face and Gesture*, pages 423–428, 2002.

[3] C.-C. Chang, J.-J. Chen, W.-K. Tai, and C.-C. Han. New approach for static gesture recognition. *Journal of Information Science and Engineering*, 22(5):1047–1057, September 2006.

[4] W. Freeman and M. Roth. Orientation histograms for hand gesture recognition. In *International Workshop on Automatic Face and Gesture Recognition*, pages 296–301, 1994.

[5] J. Segen and S. Kumar. Fast and accurate 3d gesture recognition interface. In *ICPR '98: Proceedings of the 14th International Conference on Pattern Recognition-Volume 1*, page 86, Washington, DC, USA, 1998. IEEE Computer Society.

[6] T. Starner and A. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *Pattern Analysis and Machine Intelligence, IEEE Transactions*, 20(12):1371–1375, December 1998.

[7] S. G. Wysoski, M. V. Lamar, S. Kuroyanagi, and A. Iwata. A rotation invariant approach on static-gesture recognition using boundary histograms and neural networks. In *Proceedings of the 9th*

*International Conference on Neural Information Processing (ICONIP*, pages 2137–2141, 2002.

[8] H. Zhou, D. Lin, and T. Huang. Static hand gesture recognition based on local orientation histogram feature distribution model. In *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 10*, page 161, Washington, DC, USA, 2004. IEEE Computer Society.