IDIAP RESEARCH REPORT

Hand Posture Classification and Recognition using the Modified Census Transform

Agnès Just [a] Yann Rodriguez [a]
Sébastien Marcel [a]

IDIAP–RR 06-02

January 2006

to appear in
Proc. of the IEEE Int. Conf. on Automatic Face and Gesture Recognition 2006

[a] IDIAP Research Institute

IDIAP Research Report 06-02

Hand Posture Classification and Recognition using the Modified Census Transform

Agnès Just Yann Rodriguez Sébastien Marcel

January 2006

to appear in
Proc. of the IEEE Int. Conf. on Automatic Face and Gesture Recognition 2006

Abstract. Developing new techniques for human-computer interaction is very challenging. Vision-based techniques have the advantage of being unobtrusive and hands are a natural device that can be used for more intuitive interfaces. But in order to use hands for interaction, it is necessary to be able to recognize them in images.

In this paper, we propose to apply to the hand posture classification and recognition tasks an approach that has been successfully used for face detection [3]. The features are based on the Modified Census Transform and are illumination invariant. For the classification and recognition processes, a simple linear classifier is trained, using a set of feature lookup-tables. The database used for the experiments is a benchmark database in the field of posture recognition. Two protocols have been defined. We provide results following these two protocols for both the classification and recognition tasks. Results are very encouraging.

1 Introduction

Computers are a key element of our society. They are well integrated in our everyday life. We use them to write documents, send and receive e-mails, but also to store and retrieve data. Many computer applications require more and more Human-Computer Interaction (HCI). Nowadays, HCI is usually done using dedicated devices such as mice, keyboards and graphic boards. But these interaction ways are not natural nor intuitive. The easiest way to interact with the machine would be to use means of communication available in human-to-human communication. These means can be speech, facial expression and also body language/gestures. Transforming hands into a device for computer-human interaction is a way to make the computers easiest to communicate with. In our day-to-day life, our hands play an important role in communication. They are a type of communication very rich in many ways: we use them to point at objects, express ideas ('stop', 'OK sign', etc). Integrating the use of hands in HCI would be of great benefit for the user.

In order to use the hands for HCI, it is necessary to provide the means by which they can be interpreted by the computer. A common technique is to instrument the hand, using magnetic sensors, acoustic or inertial trackers. An other possibility is to use gloves such as the CyberGlove®[1]. But these interaction techniques, even giving accurate results, are not very natural, as the gesturer's hand is adorned. The less intrusive techniques for that purpose are of vision techniques. The interaction is more natural when no device interferes in the interaction process. Furthermore, vision-based approach carry the advantage of being unobtrusive.

Hands are one of the most natural device. If we want to use them as such, we have to distinguish between "hand gesture" and "hand posture": dynamic versus static. A hand gesture can be the trajectory of the hand or a sequence of hand poses. On the contrary, a hand posture is defined by the pose or configuration of the hand in one single image.

In this paper, we will focus on the problems of hand posture classification and recognition. For that purpose, we are using a benchmark database, namely the Triesch Hand Posture Database [7]. This database contains images of ten hand postures against cluttered and uncluttered background. In order to recognize these hand postures, we will use the approach taken by Fröba and Ernst for face detection, using the Modified Census Transform (MCT) [3].

The rest of the paper is organized as follows. Section 2 briefly reviews some related works. In section 3, we will then describe our proposed approach and introduce the MCT feature extraction method. Section 5 provides experiment results for both hand posture classification and recognition on the Triesch database following two different protocols. Finally, we will conclude and propose some future research directions.

2 Related Work

Static gesture recognition or hand posture recognition is a pattern recognition problem. The first step before using any standard pattern recognition technique is the feature extraction step. Features correspond to the most discriminant information contained in the recorded image. It has to deal with the problem of cluttered/uncluttered background and changes in lighting conditions. It is possible to recognize hand posture by extracting some geometric features such as fingertips, finger directions and hand contours, but such features are not always available due to self-occlusion and lighting conditions. Many other non-geometric features exist such as color, and silhouette.

Concerning the recognition, Triesch and Malsburg [7] employ the *elastic graph matching* technique to classify hand postures against cluttered backgrounds. Hand postures are represented by labeled graphs with an underlying two-dimensional topology. This approach can also achieve scale-invariant and user-independent recognition. It does not need hand segmentation. But the approach is view-dependent.

[1] www.immersion.com

In [9], Marcel uses a neural network approach to classify hand postures against different backgrounds (cluttered and uniform). A model is trained for each posture. The structure of each model was experimentally chosen. Neural networks are also used by Gutta et al. [4]. They build a hybrid architecture which consists of an ensemble of connectionist networks (radial basis functions) and inductive decision trees (such as C4.5).

Another approach is taken in [1]. Hand configurations were classified in a space defined by a principal component analysis of the distribution of hand images.

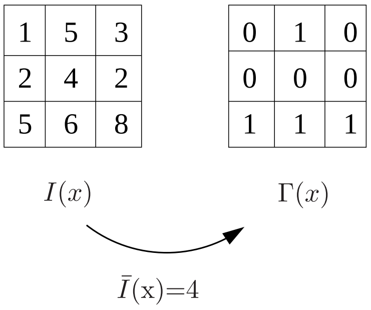
In their paper, Kölsch and Turk [8] propose to use a learning-based object detection method that was proposed by Viola and Jones [6] and primarily applied to face detection. The feature are based on the Haar basis functions and a cascade is used. The cascade consists of stages with increasing number of classifiers. Every stage of the classifier has to classify the image as positive or reject it. These layers are here to speed up the computing process. The cascade technique used in this paper is very close to the one proposed by Fröba and Ernst [3].

3 Proposed Approach

Our approach is similar to the one introduced by Fröba and Ernst [3] for the face detection task. The approach is based on a local non-parametric pixel operator: the Modified Census Transform (MCT). In order to recognize the hand postures, one classifier is trained for each posture class. A boosting procedure is both used for feature selection and classifier training.

3.1 Feature Space

The feature space is defined as a set of 3x3 kernels which emphasize the local spatial structure of an image. The Modified Census Transform (MCT) is used to compute the index of the kernels. MCT is a non-parametric transform inspired by the Census Transform first introduced by Zabih and Woodfill [10] in the context of texture analysis. MCT consists of an ordered set of binary comparison of pixel intensities between all the pixel of the 3x3 neighborhood and the mean intensity of all the pixel of the neighborhood (Figure 1).



1	5	3
2	4	2
5	6	8

0	1	0
0	0	0
1	1	1

Figure 1: Example of the Modified Census Transform.

Let $\mathscr{N}(x)$ be a 3x3 local spatial neighborhood of the pixel x such that $x \in \mathscr{N}$, and $\bar{I}(x)$ be the intensity mean of the pixel intensities of this neighborhood. The MCT generates a bit string representing which pixels in $\mathscr{N}(x)$ have an intensity lower than $\bar{I}(x)$. If $\zeta(\bar{I}(x)), I(y)) = 1$ if $\bar{I}(x) < I(y)$ is the comparison function and $\otimes$ is the concatenation operation, then the census transform is defined as:

$$\Gamma(x) = \otimes_{y \in \mathscr{N}} \zeta(\bar{I}(x), I(y)).$$

By definition, MCT is unaffected by any monotonic gray-scale transformation which preserves the pixel intensity order in a local neighborhood. We finally point out that, approximately in the same time the MCT was introduced by Fröba and Ernst [3], Jin et al. [5] proposed a very similar local structure feature. This feature, called *Improved Local Binary Pattern* (ILBP), also maps the local neighborhood surrounding a pixel. With respect to MCT, the ILBP feature only differs by the order of the bit string.

3.2 Classifier

In order to recognize the hand postures $P = \{p_i\}_{i=1}^{10}$ (10 different postures), one classifier H_i is trained for each posture class p_i. For each posture, the process can be seen as a two-class classification task. A variant of AdaBoost [2] is both used to select relevant features and train the classifier. The goal of the AdaBoost algorithm is to combine simple classifiers into a stronger one. The global classifier $H_i(\Gamma)$ is a linear combination of the selected weak classifiers $h_n(x)$.

$$H_i(\Gamma) = \sum_{n=1}^{N} h_n(\Gamma(x_n)),$$

where N is the number of weak classifiers. Each classifier h_n consists of a look-up table of size 511 (number of possible kernel indices of MCT), associated to a pixel location x_n in the image. Each bin of the look-up table contains a weight for the corresponding kernel index. The output of a weak classifier h_n associated to pixel location x_n is then the weight addressed by the kernel index computed with the MCT at pixel x_n.

To train our *two-class* strong classifiers H_i, the training data is composed of two sets: one containing sample images of the posture to recognize and another one containing sample images of *non*-posture. For the second set, we randomly collected images on the Internet and added some images of the nine remaining postures. We expect this addition to increase the robustness of the model.

The interested reader would refer to the article of Fröba and Ernst [3] for implementation details. Note that instead of the proposed four-stage cascade structure, we used only a one-stage classifier of $N = 500$ look-up tables, trained with 2500 iteration of boosting. Classification is done according to:

$$H(\Gamma) \leq T$$

where the decision threshold T is chosen on a separate validation set, by minimizing the classification error rate.

4 Experiment Set-up

4.1 Database and Protocols

The database used in this article is the Jochen Triesch database [2] which is a benchmark database in the field of hand posture recognition. It consists of 10 hand signs (cf. figure 2) performed by 24 different gesturers against different backgrounds.

The backgrounds are of three types: uniform light, uniform dark and complex (cf. figure 3). Amongst the 720 images, two were lost by Triesch.

[2] http://www-prima.inrialpes.fr/FGnet/data/09-Pets2002/data/POSTURE/

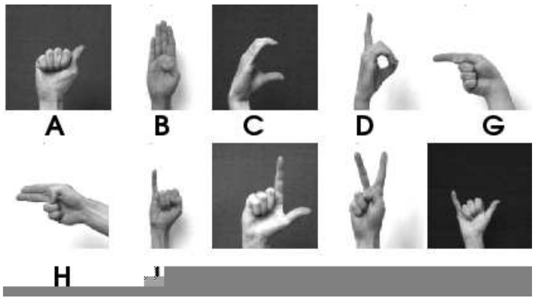


Figure 2: The 10 postures to recognize.

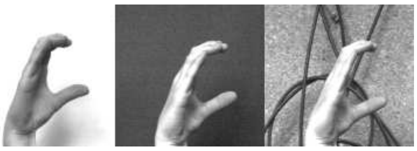


Figure 3: Three types of backgrounds.

For experiment purposes, the database has been divided into three subsets: train set T, validation set V and test set Te. The decomposition into subsets has been done following two different protocols.

In both cases, images of four people were used for the training set, and images of four other people were kept for the validation case. The images of the sixteen remaining people formed the test set. In the first case (Protocol 1), only the images against uniform background have been used for training and validation, the remaining images against complex background have been added to the test set. In the second case (Protocol 2), both complex and uniform background images have been used for training and validation. Table 1 summarizes the decomposition into the three subsets.

	Protocol 1			Protocol 2		
	T	V	Te	T	V	Te
number of people	4	4	16	4	4	16
number of images	80	80	558	119	120	479
background type	U	U	U/C	U/C	U/C	U/C

Table 1: Subsets statistics (U=uniform, C=complex).

4.2 Preprocessing

Before using the images to train and test the model, they have first been cropped to the 30×30 size followed by an histogram normalization. In order to increase the available number of images for the training and validation process, some perturbations have been added to the initial images. Three types of perturbations have been generated. The images have been shifted, scaled and rotated of a few pixels and degrees. For each original images, 30 perturbed images have been generated At the end of the process, for the first protocol, we have $4 \times 2 \times (30 + 1) = 248$ images per posture for the training and validation set. For the second protocol ,we have $4 \times 3 \times (30 + 1) = 372$ images per posture, except for the 'H' posture for which we only have 351 images (one image of the initial database was lost).

In the case of the test set, no perturbation has been added to the images. The test set contains the initial images cropped to the 30 × 30 size.

5 Results

This section is divided in two parts. In the first one, we consider each hand posture separately and are interested in posture classification. In the second part, after having shown the validity of our approach to perform correct classification, we consider the task of hand posture recognition.

5.1 Hand Posture Classification

First of all, we would like to verify that for each hand posture p_i, our two-class model H_i is able to correctly classify the test postures p_i. For that purpose, we test the model H_i on the test images of posture p_i and report the classification rate on table 2.

	Uniform Background		Complex Background	
	Protocol1	Protocol2	Protocol1	Protocol2
A	100	100	91.67	100
B	93.75	100	75	100
C	96.88	100	66.67	93.75
D	100	100	87.5	100
G	100	100	87.5	100
H	100	100	100	100
I	100	100	95.83	93.75
L	100	100	100	100
V	96.77	100	54.17	100
Y	96.88	100	62.5	87.5
average	98.4	100	82.1	97.5

Table 2: Classification rate (in %) on the test set

We notice that our models H correctly classify most of the hand postures. Regarding table 2, some remarks can be drawn.

1. **Background**: We observe that for both background conditions, the classification rate is high. The average classification rate is equal to 99.2% for the uniform background, and 89.8% for cluttered conditions. As expected, classification rate with uniform background provides better results than with complex background.

2. **Posture**: Against uniform background, all postures are well classified, whatever the protocol we use. Concerning the complex background, results depend on the protocol. But we remark that

some postures are more difficult to classify. The 'Y' posture for instance achieves an average classification rate of 75%.

3. **Protocol**: The difference between the two protocols lies in the composition of the training and validation sets. In Protocol 1, the training and validation sets contain no images against complex background. In Protocol 2, those images are included to match the testing conditions. As stated above, for all hand postures and both protocols, the classification rates are very high for the uniform background. In the cluttered conditions, for Protocol 2, results are as good as in the uncluttered conditions. On the other hand, for Protocol 1, the recognition rate decreases for most of the postures. The difference in the performances is particularly obvious with the 'C', 'V' and 'Y' postures. It seems that using the images against complex background in the training process adds some relevant information. One explanation could be found in the choice of the pixel locations for the classifiers. The choice of theses pixels is even more restricted due to the presence of a changing background. Then the boosting algorithm has to focus much more on the pixels contained in the hand postures. Thus, the classifier is able to recognize more accurately the postures in cluttered conditions.

5.2 Hand Posture Recognition

In the previous section, each posture p_i was considered independently. We verified that the test images of posture p_i were correctly classified by model H_i. In this section, we are interested in the recognition task, i.e. given a unknown posture, we would like to find its posture class label. For that purpose, we chose a "one versus all" strategy. For a given posture test image, we apply all the models $H = \{H_i\}_{i=1}^{10}$ and consider the model giving the highest score to label the test image.

The recognition rate for each posture p_i, both for uniform and complex backgrounds, are reported in table 3.

	Uniform Background		Complex Background	
	Protocol1	Protocol2	Protocol1	Protocol2
A	100	100	100	100
B	93.75	93.75	93.75	93.75
C	93.75	93.75	75	93.75
D	93.75	84.38	62.5	81.25
G	96.88	100	50	68.75
H	84.38	90.63	87.5	87.5
I	84.38	90.63	56.25	62.5
L	84.38	96.88	37.5	75
V	87.10	96.77	56.25	87.5
Y	81.25	81.25	25	62.5
average	89.97	92.79	64.38	81.25

Table 3: Classification rate (in %) on the test set

Several remarks can be drawn from this table.

1. **Background**: In general, the recognition rate is higher for the images against uniform than complex background. We notice that some postures are not affected by the background type such as 'A' or 'B', while other postures are strongly affected, such as 'G', 'I', 'L', 'V' or 'Y'. The common features of these postures is a closed fist with one ore two single pointing fingers. The detection of these thin finger regions seem to be very sensitive to the background type. In other words, they are “sunk” in the background and thus difficult to find out.

2. **Posture**: Some postures, for both background types, seem to be easier to recognize. The 'A' posture for instance, achieves 100% recognition in both background conditions. On the other hand, the 'Y' posture achieves the lowest recognition rate in both conditions. The explanation may be found in the high variability of the hand posture shape. While the 'B' posture will be performed in a similar manner by every gesturer, it will not be the case with the 'Y' posture.

3. **Protocol**: We can notice that integrating images against complex background in the training and validation sets helps the algorithm to model more accurately the postures, even if some postures seem to be particularly difficult to learn. It is the case for the 'D', 'G', 'I', 'L' and 'Y' postures.

6 Conclusions and Future Directions

In this paper, it has been shown that the boosting approach based on MCT features, applied with success to face detection, can also be applied to the problem of hand posture recognition. We have shown the importance of adding images against complex background in the training set. This leads to a significant improvement of the recognition rate in complex background conditions.

The results are really encouraging but there are still some postures that are more difficult to recognize such as the 'G', 'I' and 'Y' postures. These postures have in common a fist conformation with one or two single pointing fingers (thin regions). It makes them difficult to recognize against complex background as the fingers are lost in the background. On possible way to avoid this problem would be to increase the size of the kernels in order to catch more information around the fingers.

Another problem lies in the fact that we have only used cropped images. In each image, the posture is perfectly centered and all images have the same size. But in the real life we do not have already cropped images of hand postures, and hands can be of different size, depending on the depth in the image. Most of the time, the hand is only a part of the image. To overcome this limitation, we are currently working on hand posture detection using a *sliding window* technique. This approach, widely employed in face detection, consists of scanning an image at different positions and scales. The final goal is to build a fully automatic hand posture recognition system.

7 Acknowledgments

This research has been carried out in the framework of the Swiss National Science Foundation through the National Center of Competence in Research (NCCR) on “Interactive Multimodal Information Management (IM2)”. This work was also cofunded by France Telecom R&D through the GHOST project (project number 021B276).

References

[1] J.L Crowley and J. Martin. Visual processes for tracking and recognition of hand gestures. In *Workshop on Perceptual User Interfaces (PUI'97)*, October 1997.

[2] Y. Freund and R. E. Shapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, pages 771–780, 1999.

[3] B. Froba and A. Ernst. Face detection with the modified census transform. In *Proceedings of the Automatic Face and Gesture Recognition Conference*, 2004.

[4] S. Gutta, J. Huang, I.F. Imam, and H. Wechsler. Face and hand gesture recognition using hybrid classifiers. Technical report, 1996.

[5] H. Jin, Q. Liu, H. Lu, and X. Tong. Face detection using improved LBP under bayesian framework. In *Proceedings of the Third International Conference on Image and Graphics (ICIG), Hong Kong, China*, pages 306–309, 2004.

[6] M. Jones and P. Viola. Fast multi-view face detection. Technical Report TR2003-96, MERL, July 2003.

[7] J.Triesch and C.von der Malsburg. Robust classification of hand postures against complex backgrounds. In IEEE Computer Society Press, editor, *Proceedings of the Automatic Face and Gesture Recognition Conference*, pages 170–175, Killington, Vermont, USA, October 14-16 1996.

[8] M. Kölsch and M. Turk. Robust hand detection. In *Proceedings of the Automatic Face and Gesture Recognition Conference*, pages 614–619, 2004.

[9] Sébastien Marcel. Hand posture recognition in a body-face centered space. In *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*, pages 302–303, New York, NY, USA, 1999. ACM Press.

[10] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *Proceedings of the European Conference on Computer Vision*, pages 151–158, 1994.