15th International Symposium on Computer Methods in Biomechanics and Biomedical Engineering
and
3rd Conference on Imaging and Visualization
CMBBE 2018
P. R. Fernandes and J. M. Tavares (Editors)

CONTACTLESS HAND IDENTIFICATION USING MACHINE LEARNING

Armin Dietz*, **Joachim Hienzsch***, **Eduard Reithmeier***

*Leibniz Universität Hannover, Institute of Measurement and Automatic Control
Nienburger Str. 17, Hannover, Germany
armin.dietz@imr.uni-hannover.de

Keywords: Biometrics, Machine Learning, Hand Identification, User Authentication, Human Machine Interfaces.

Abstract: *Hand gesture recognition systems are becoming more popular for human machine interfaces (HMI) in consumer devices, as well as in industrial and medical applications. Therefore, it is desirable to control access rights by users or user groups. We propose a user identification system that is able to recognize users by their hands. Two concepts are presented, that vary in the degree of required user cooperation. A depth image of a time-of-flight sensor is used to detect hands and to segment them from the background. After transformations on the region of interest, an infrared image of the same sensor serves as input to a convolutional neural network for classification. Experimental results indicate the feasibility of the user identification system, with rates above 98% of success in classification of up to 22 users.*

1. INTRODUCTION

Biometrics authentication is used for access control and user identification. It relies on the recognition of physiological or behavioral characteristics of individuals [1]. The gait of a person, for example, is considered as behavioral characteristic, whereas physiological characteristics include features of a face, fingerprints, or hand geometry. Hand gesture recognition system are becoming more popular for HMI in consumer as well in industrial and medical devices, e.g the operation of surgery lights by hand movements in the air [2]. One problem of these systems is that it is difficult to differentiate between the user input of several people. This can also cause security concerns, when an unauthorized user is controlling a system. Contactless hand identification could help overcome this problem. Several biometrics authentication system exist that identify a user by its hands [3, 4, 5, 6, 7]. However, to our best knowledge, none of them work contactless.

2. MATERIALS AND METHODS

To identify a person by its hands, we apply the following steps: A depth image of a time-of-flight sensor is used to detect hands and to segment them from the background. After transformation and normalization of a region of interest, an infrared image of the same sensor serves as input to a convolutional neural network for classification. A Microsoft Kinect v2 sensor is used to acquire a 512 px x 424 px resolution depth and infrared stream. The sensor is facing downwards in a slight angle from 1.9 meters height. The field of view of the sensor is 70° x 60°. The value of each pixel in a depth image represents the distance of an object to the sensor in millimeter.

2.1 Hand detection and normalization

To simplify the hand detection, it is assumed that the hand is held inside a region of 391 px x 221 px with the user's arm crossing the region's upper border from the top (see figure 1).

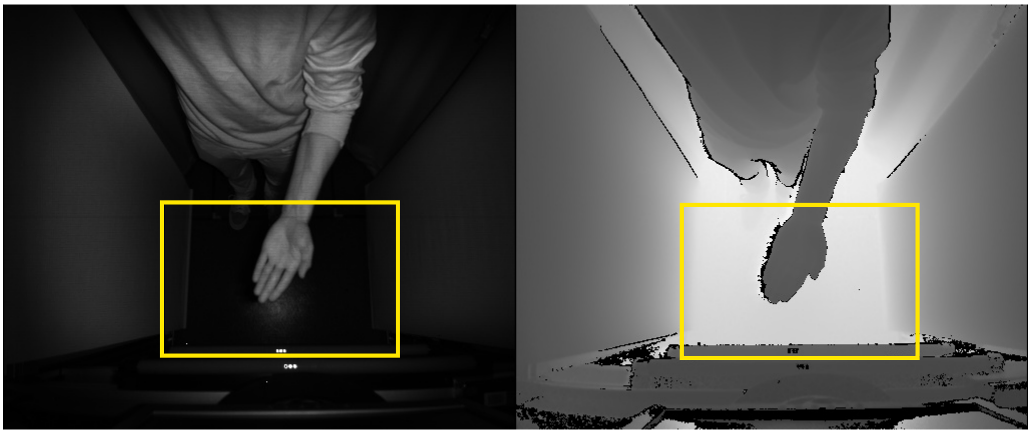

Figure 1. Raw infrared (left) and depth (right) image with corresponding region of 391 px x 221 px. The user's arm is crossing the region's upper border from the top.

Next, we segment the hand area from the background using a depth threshold, setting all depth pixels z to zero that are within:

$$z_{min} - s < z < z_{min} + s \tag{1}$$

with z_{min} being the minimum depth value of the first pixel row - that is the row where the arm is crossing the region's upper border - and s being a set value of 100 mm. To exclude any residing background pixels that are not part of the hand or arm, we apply the flood-fill algorithm with four neighboring pixels on the binarized image. The start node is the position of z_{min}, which is likely to be a part of the hand or arm. Any pixel that is not connected to the start node is considered as background and set to zero.

In a next step, we rotate the image so that the hands on all images are aligned in the same direction. We determine the contour of the hand. The point with the greatest distance to the contour is defined as the center. A second point that is below the center point and has the greatest distance from that center point is defined to be the point of the middle finger (see figure 2. Finally the image rotated so that the connecting line between those two points is aligned vertically.

Due to the field of view of the Kinect, the projected size of an object on the image plane changes depending on the distance of the object to the sensor. We determine experimentally a

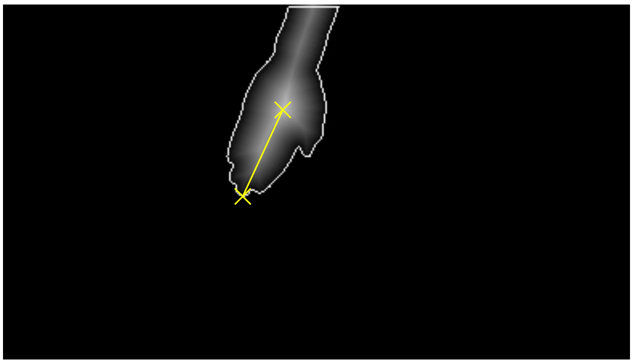

Figure 2. Center and middlefinger point with connecting line of the segmented hand image.

linear relationship of the depth value z and the projected object width, that is shown in equation 2). We measure the projected width of an object at 20 distances in between 0.6 m and 1.0 m. We use the equation to scale a depth image of a segmented hand of a subject, which achieves a consistent size of the projected hand independent of its distance to the sensor. This also allows us to use the hand size as a feature of a subject.

$$f(z) = 0.0031065671z - 0.1330244801 \tag{2}$$

After scaling the hand image, a smaller region of interest of 128 px x 128 px is defined, that mostly contains pixels of the hand. Based on the center point of the hand, a circle with the radius of the smallest distance from the center point to the contour of the hand is drawn. The image is cropped at the most upper point of the circle, 128 px below it, and 64 px to each side, as can be seen in figure 3. We only use the infrared image for our classification network. The infrared values are scaled from 0 to 1.

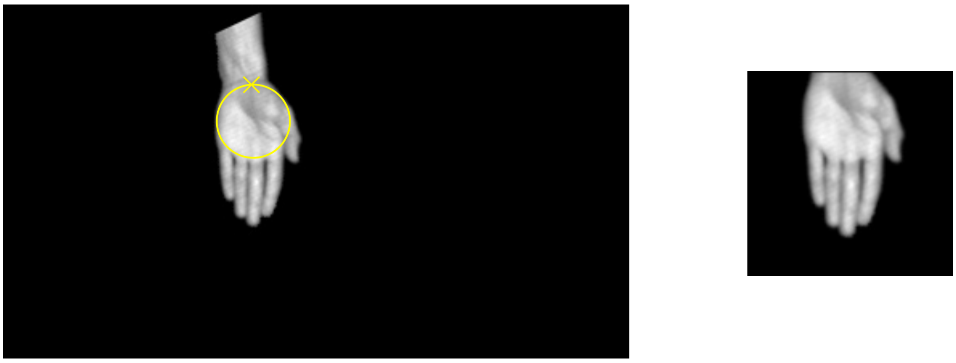

Figure 3. Rotated hand image with circle around the center point of the hand (left). The upper point of the circle is the starting point for the final crop, as seen on the right.

2.2 Architecture of convolutional neural network

We use a convolutional neural network for classification. The architecture of the network is summarized in table 1. The input to the network is a pre processed infrared image with a dimension of 128 px x 128 px. The output layer consists of 22 dense neurons with softmax activation. The number of output neurons is depending on the number of subjects used for training.

Layer type	Filter	Strides	Activation
Convolution	$32 \times 3 \times 3$	1×1	ReLU
Max pooling	2×2	1×1	
Convolution	$32 \times 3 \times 3$	1×1	ReLU
Max pooling	2×2	1×1	
Flatten			
0.25 Dropout			
Fully connected	128		ReLU
0.5 Dropout			
Fully connected	22		Softmax

Table 1. Architecture of neural network. The convolutional layers are zero padded.

2.3 Training

We trained two classification models, each with a different dataset. The datasets differ in the degrees of freedom of the hands. For dataset 1, the hand is held flat with the neighboring fingers touching each other. 8800 images of 22 subjects are recorded for dataset 1, with varying angles of the flat hands (see figure 4). Dataset 2 consists of 44264 images of the same 22 subjects. It includes the images of dataset 1 and is extended by images, where the fingers where spread out randomly, as can be seen in figure 5.

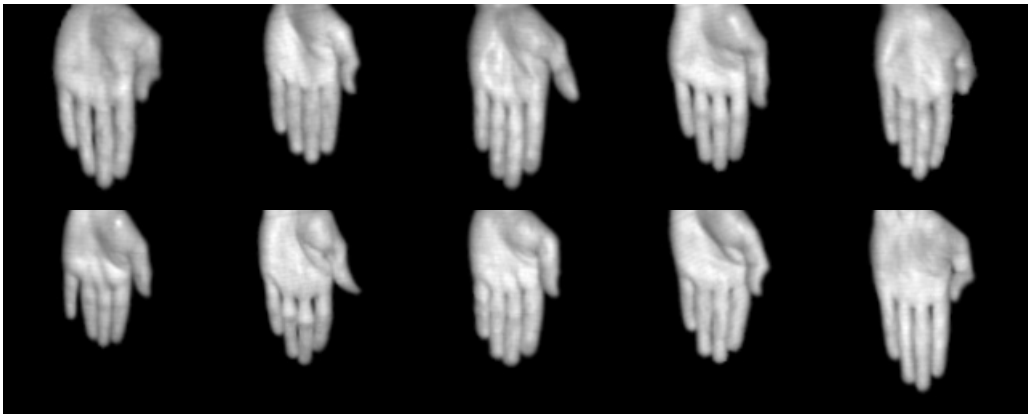

Figure 4. Hand pose in dataset 1, examples of ten different subjects.

Data augmentation is applied on both datasets to achieve a higher degree of variation. The images are randomly rotated by ± 10 degrees and compressed in horizontal and vertical direction. For the vertical compression, 0-50 random lines of background pixel are added on the bottom of the image. The image is then resized again to its original size. For horizontal compression, the image is first resized to w px x 128 px, where w is a random width between 70

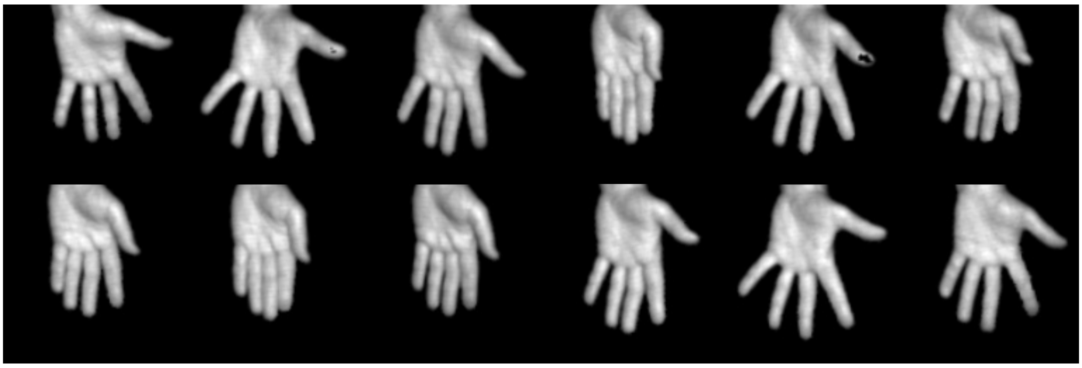

Figure 5. Examples by a single subject of allowed hand poses of dataset 2.

and 128. Missing background pixel are then added to restore the original size. The motivation of the compression is to simulate any tilting of the hand.

Both networks are trained with Adam optimization. We use a batch size of 32 and categorical crossentropy as a loss function. We use k-fold cross validation. Dataset 1 is trained for 1000 iterations with a fold of $k = 10$, resulting in 7920 training and 880 validation images. Dataset 2 is trained for 600 iterations with a fold of $k = 7$, resulting in 37940 training and 6323 validation images.

3. RESULTS

Table 2 shows the quantitative results of the experiments. It was possible to achieve accuracies above 98% with both datasets.

Dataset	No. of subjects	Accuracy
1	22	$98.91\% \pm 0.52\%$
2	22	$98.68\% \pm 0.56\%$

Table 2. Accuracies for each dataset with k-fold cross validation.

Figure 6 and figure 7 display the limits of tilting in vertical and horizontal positions and stretching or curling until which the subject is correctly classified for dataset 1. Subjects whose hand poses differ from the specified flat pose are not correctly classified.

Figure 8 shows that the hands can be tilted further in dataset 2 and still be classified correctly. However, 9 displays cases where the hand poses differ too much from the specified pose and are not classified correctly.

4. DISCUSSION

The hand identification with dataset 1 achieves high accuracies. By demanding a flat hand position with touching neighboring fingers, low degrees of variance are achieved. Additionally, the pose is easy to execute. However, a wrong execution by a subject leads likely to a wrong classification. Therefore, a high degree of user cooperation is required. A disadvantage of this concept is that recorded images of a subject are very similar. Classification can completely

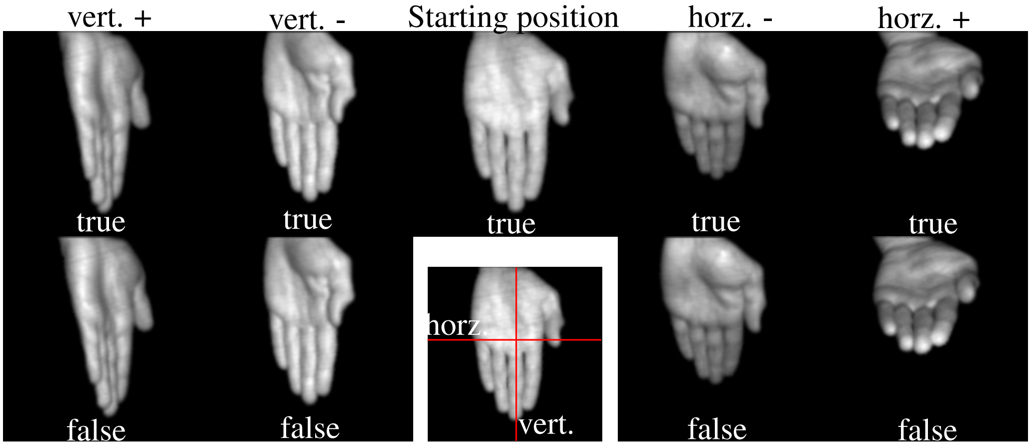

Figure 6. Examples of correctly and incorrectly classified hand poses of dataset 1. The top row displays the last possible tilting position before the subject was not classified correctly anymore. The bottom row displays the first first falsely classified subject when tilting too far.

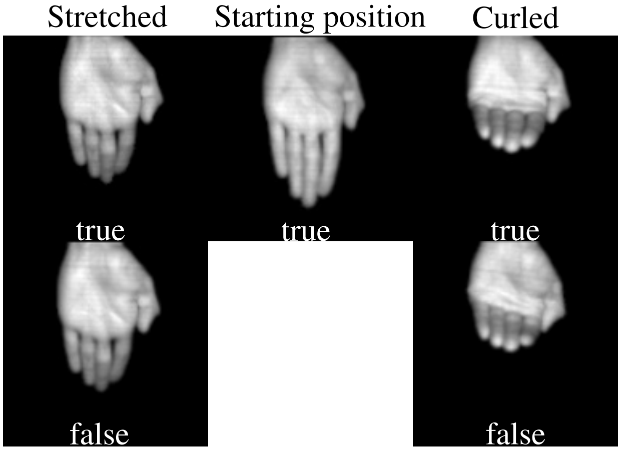

Figure 7. Examples of correctly and incorrectly classified hand poses of dataset 1. The top row displays the poses of the last correct classification when a hand is stretched or curled. The bottom row displays the poses of the first incorrect classification.

fail if the pose is falsely executed during training. Higher accuracies and robustness could be achieved by recording greater tilting angles.

In the second concept, the degrees of freedom for possible hand poses are increased. By increasing the variation of hand poses in the training data set, the user identification is also successful in a higher variation of poses. Although less likely to occur than in concept 1, a high deviation of the required pose can also lead to a wrong user identification.

In both cases, it is important to record training data with a high variance of poses from the same person to ensure that the users are correctly identified in case their hand poses differ in later executions.

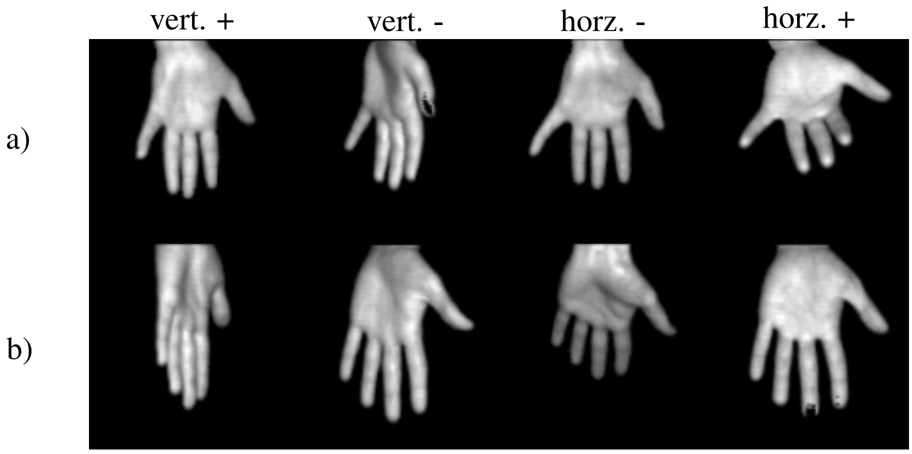

Figure 8. Examples of correctly classified hand poses of dataset 2. a) and b) are hand images of two different subjects.

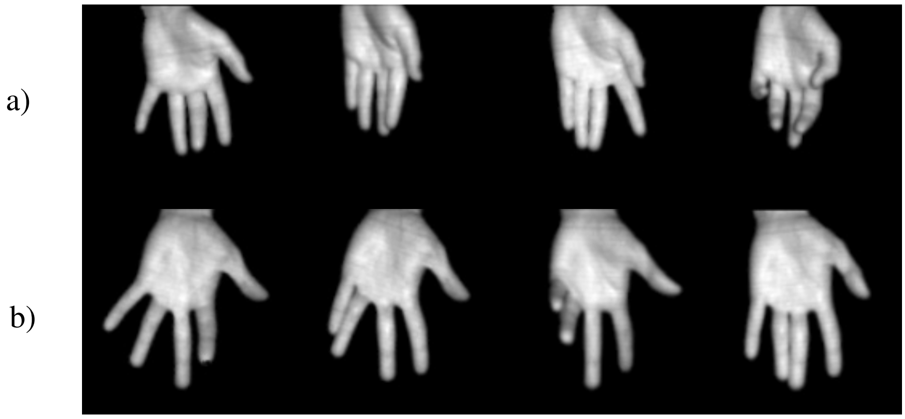

Figure 9. Examples of hand poses of dataset 2 that were not correctly classified. a) and b) are hand images of two different subjects. Note that these poses differ vastly from the original pose with a flat hand and evenly spread out fingers.

5. CONCLUSIONS

We presented an approach to contactlessly identify users by their hands using a time-of-flight sensor and machine learning. Two concepts are presented, that vary in the degree of required user cooperation. Both of them achieve a high accuracy with more than 98% of correct identifications of 22 subjects. We show that contactless user identification is feasible when a high variance of poses are recored as training data.

References

[1] A. K. Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *IEEE Trans. Cir. and Sys. for Video Technol.* **14**, 1 (January 2004), 4-20. DOI=http://dx.doi.org/10.1109/TCSVT.2003.818349

[2] A. Dietz, S. Schröder, A. Pösch, K. Frank and E. Reithmeier. 2016. Contactless Surgery Light Control based on 3D Gesture Recognition. *EPiC Series in Computing*, **40**,138-146, 2016 DOI=https://doi.org/10.29007/zmz9

[3] A. K. Jain and N. Duta. Deformable matching of hand shapes for user verification. *Image Processing, ICIP 99. Proceedings. 1999 International Conference on. Bd. 2. IEEE.* 857-861, 1999.

[4] C. Oden, A. Ercil and B. Buke. Combining implicit polynomials and geometric features for hand recognition. *Pattern Recognition Letters*, **24.13**, 2145-2152, 2003.

[5] A. Ross, A. Jain and S. Pankati. A prototype hand geometry-based verification system. *Proceedings of 2nd conference on audio and video based biometric person authentication.*, 166-171, 1999.

[6] R. Sanchez-Reillo, C. Sanchez-Avila and A. Gonzalez-Marcos. Biometric identification through hand geometry measurements. *IEEE Transactions on pattern analysis and machine intelligence* **22.10**, 1168-1171, 2000.

[7] E. Yoruk et al. Shape-based hand recognition. *IEEE transactions on image processing* **15.7**,1803-1815, 2006.