Contactless biometric hand geometry recognition using a low-cost 3D camera

Jan Svoboda, Michael M. Bronstein
Faculty of Informatics
Universita della Svizzera Italiana
Lugano, Switzerland
{jan.svoboda,michael.bronstein}@usi.ch

Martin Drahansky
Faculty of Information Technology
Brno University of Technology
Brno, Czech republic
drahan@fit.vutbr.cz

Abstract

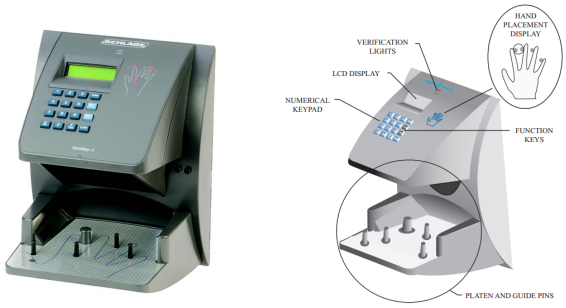
In the past decade, the interest in using 3D data for biometric person authentication has increased significantly, propelled by the availability of affordable 3D sensors. The adoption of 3D features has been especially successful in face recognition applications, leading to several commercial 3D face recognition products. In other biometric modalities such as hand recognition, several studies have shown the potential advantage of using 3D geometric information, however, no commercial-grade systems are currently available. In this paper, we present a contactless 3D hand recognition system based on the novel Intel RealSense camera, the first mass-produced embeddable 3D sensor. The small form factor and low cost make this sensor especially appealing for commercial biometric applications, however, they come at the price of lower resolution compared to more expensive 3D scanners used in previous research. We analyze the robustness of several existing 2D and 3D features that can be extracted from the images captured by the RealSense camera and study the use of metric learning for their fusion.

In the past decade, 3D sensors have experienced a series of revolutionary technological breakthroughs, shrinking in size and dropping in price by several orders of magnitude. The new generation of low-cost small form-factor sensors manufactured by companies such as PrimeSense, SoftKinetic, and Intel has enabled a wide range of commercially viable applications of 3D computer vision. A particularly appealing application field of 3D sensors is biometric person recognition, as there is a strong experimental evidence that 3D information allows improving the accuracy and robustness of biometrics. In particular, the valuable 3D information is contained mainly in human ear [5], face [2, 3], fingers of the hand [16, 22] and also the whole hand geometry and palmprint [9, 10].

However, with a few recent exceptions, most of the academic research works on 3D biometrics rely on expensive, large, and slow 3D sensors that provide very high-quality scan but at the same time are impractical in commercial applications due to high cost and size (for example, Minolta Vivid 910 used in [22] has a resolution of 0.1mm but costs about $30K and weights 15kg). Furthermore, the research on using 3D biometric features has been mostly focused on face recognition, leading to several commercial 3D face recognition systems. Other biometric modalities such as 3D hand recognition have received relatively less attention, and we are unaware of any commercially successful product based on 3D hand biometrics currently on the market.

The main goal of our paper is to fill this gap. We explore the the possibility of using cheap small form-factor 3D sensors (in particular, we used the recently announced Intel RealSense near-infrared coded-light 3D camera, which is 3.5mm thick and costs about 30$) for contactless 3D hand biometric recognition using existing 2D and 3D hand geometry features. The small dimensions and price of the RealSense 3D camera comes at the expense of lower image quality compared to more expensive larger 3D sensors, requiring a special positioning loop guiding the user during the acquisition phase to ensure a high-quality capture, as well as the use of metric learning for feature comparison. We scan the dorsal part of the hand, and do not use the palmprint information (our approach utilizes purely geometric features of the hand surface). The result of our work is a small, inexpensive, and practical 3D hand biometric recognition system that works in real time on a standard PC and achieves performance comparable to that of state-of-the-art systems reported in the literature at a tiny fraction of their cost and size.

The rest of the paper is organized as follows. In Section 1, we overview prior works on 3D biometrics focusing on hand biometrics. In Section 2, we describe our system, including the automatic hand positioning loop and feature extraction and matching algorithms. Section 3 provides the evaluation of our system on a proprietary database of 100 individuals hands acquired using the Intel RealSense camera. Finally, Section 4 concludes the paper.



(a) (b)

Figure 1. Existing commercial solution: (a) HandKey II system; (b) HandKey II schematics.

1. Background

A typical 2D hand recognition system (Figure 1) requires the user to place the hand on a reflective flat surface in a specific position constrained with special pegs built into the surface as introduced in [8] and [7]. Such an approach lowers the error rate of the system at the expense of burdening the user with proper hand placement. Peg-free 2D hand recognition systems accounting for different hand rotations have been proposed using palmprint [12], palmprint and geometry [14], finger knuckle [13], eigenpalm and eigenfingers features [17] or a model-guided approach [23]. However, most of these methods still require the user to place the hand on a flat surface, impacting the form-factor of the system and raising hygiene concerns.

Towards the goal of making contactless hand recognition systems, the use of 3D data has been proposed in [16, 22] using only finger surface, in [9, 10] using both 2D and 3D hand geometry and palmprint or in [24] using the 3D palmprint information exclusively.

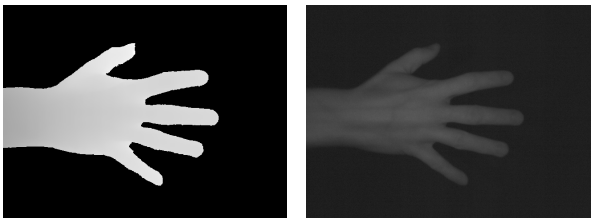
While showing the possibility of contactless systems and a potential of significant accuracy improvements, these works were done in a laboratory environment, using large and expensive 3D sensing devices that provide very accurate scans but at the same time are commercially impractical. Similar hardware was also used in other biometric modalities such as 3D face [4, 2, 3] and 3D ear recognition [5], though some very recent papers studied the use of Microsoft Kinect sensor [15, 6].

2. Proposed solution

2.1. Sensor

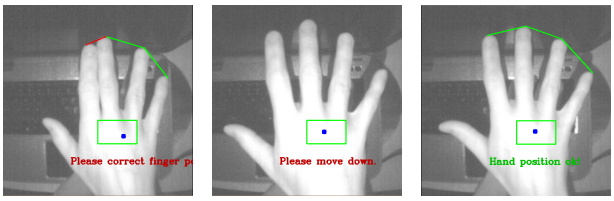
Our system uses the Intel RealSense 3D camera [1] (Figure 2), a time-multiplexed structured light sensor inferring the depth information by projecting a sequence of infrared code patterns onto the object surface. The output of the RealSense camera is a video of up to 60 VGA (640×480) depth frames per second, and a synchronized stream of VGA IR frames. There is a pixel-wise correspondence between the depth and IR frames (Figure 3).

Figure 2. Intel RealSense camera.



(a) (b)

Figure 3. Intel RealSense output: (a) Depth image; (b) IR image.



(a) (b) (c)

Figure 4. Positioning loop examples: (a) hand with index and middle finger too close to each other; (b) hand too close to the sensor; (c) properly positioned hand. The user positions the hand with the dorsal side facing the camera following the feedback provided by the positioning loop in real time. Once the hand is positioned properly, a 3D frame is automatically captured.

2.2. Hand positioning loop

As our system is completely contactless, there are no physical constraints on the position of the user's hand, as opposed to traditional hand biometric devices in which the user is asked to position the hand on a surface guided by protruding pegs. The "hard" hand positioning is replaced by the positioning look depicted in Figure 4, guiding the user how to properly position the hand and ensuring that the hand is captured only when in appropriate position.

The positioning loop extracts a set of features (described in details in the following section) from the 3D video captured by the sensor and verifies that a set of five positioning constraints is satisfied. The *depth constraint* ensures that the hand is within a range of ideal distances from the camera. *Centering constraint* restricts the x- and y- coordinates of the hand barycenter to the central region of the image. *Finger count constraint* ensures that the detected hand has exactly five fingertips (for simplicity, we do not consider the

case of bent fingers). *Fingertip distance constraint* ensures that the neighboring fingers are sufficiently apart, ensuring that the hand silhouette is fully visible. Finally, the *rotation constraint* ensures that the hand is positioned approximately in the camera image plane. The computation of the positioning constraints is lightweight, allowing to run the positioning loop in real time on a low-power platform.

2.3. Feature extraction

The hand image captured by the RealSense camera undergoes several pre-processing steps. First, we perform foreground segmentation by simply thresholding the depth image values within a fixed interval of approximately 20–30 cm, revealing a silhouette of the hand. Second, the hand barycenter is computed by applying the distance transform on the hand silhouette [19]. Third, the contour of the hand is extracted using the algorithm described in [20].

Using the hand contour, we detect the *fingertips* and *finger valleys* depicted in Figure 5(a) in green and red, respectively. Fingertips are detected as local maxima of the curvature along the hand contour, with additional distance and orientation constraints preventing the detection of outliers. Finger valleys are detected in a similar way, as four points with the highest curvature lying between pairs of adjacent fingertips. Those points are, however, not enough. In order to compute the finger axis, each finger needs one fingertip and two finger valleys assigned. To obtain exactly two finger valley points for each finger, additional 'artificial' finger valleys have to be computed for the little finger and thumb by connecting the detected finger valleys (red lines in Figure 5(a)) and finding their intersection with the hand silhouette. Next, using fingertips and finger valleys, we compute the *axis* of each finger. Then, we compute the *wrist line* (Figure 5(c)) as a line perpendicular to the line connecting the middle fingertip and the hand barycenter at a fixed distance from the barycenter.

Finally, we define two feature vectors of 2D (Figure 7(a)) and 3D (Figure 7(b)) geometric measurements on the hand. The 2D feature vector is 41-dimensional and contains *finger length* (Figure 6(a)), *finger width* (measured at several points along the finger, Figure 6(c)), and *wrist to valley* distances (Figure 6(e)), all computed using the Euclidean metric. The 3D feature vector is 137-dimensional and contains finger widths measured by traversing the 3D hand surface (Figure 6(g)) and *axis to surface distances* (Figure 6(f)). s Both 2D and especially 3D feature vectors may suffer from their higher dimensionality. This problem can be however easily overcome by using similarity-sensitive hashing methods described in [18].

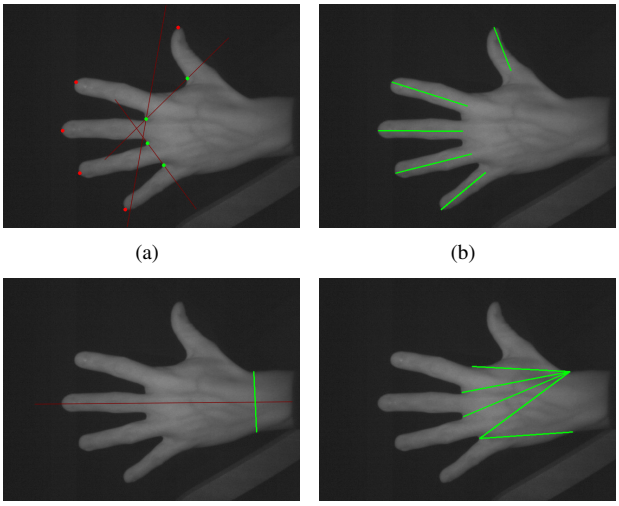


Figure 5. Processing of the hand surface: (a) detected fingertips (red dots) and finger valleys (greed dots); (b) computed finger axes (green lines); (c) extracted wrist line; (d) computed wrist to valley distances.

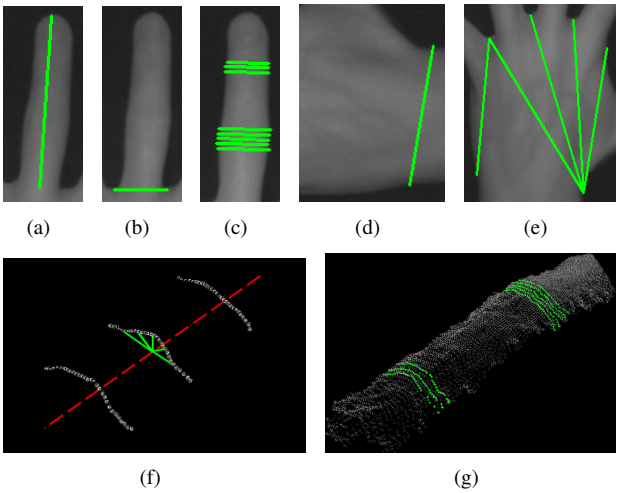


Figure 6. 2D features: (a) finger length; (b) finger valley distance; (c) finger widths; (d) wrist width; (e) wrist to valley distances. 3D features: (f) finger axis-surface distance (green lines on the middle segment); (g) finger widths (green segments).

2.4. Feature matching

The 2D and 3D feature vectors are matched separately using the *Mahalanobis distance*

$$d_A(x, y) = \left((x - y)^\top A (x - y)\right)^{1/2}, \qquad (1)$$

where x, y are the n-dimensional feature vectors ($n = 41$ and 137 for 2D and 3D features, respectively), and A is an $n \times n$ positive-semidefinite matrix defining the weight of

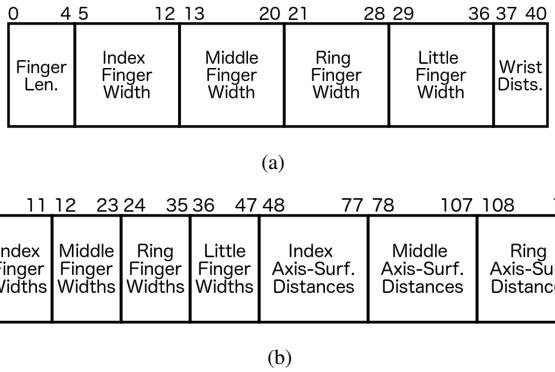


Figure 7. Feature vectors used for matching: (a) 2D feature vector - features extracted from 2D image; (b) 3D feature vector - features extracted from 3D point cloud.

different features. We employ *large margin nearest neighbors* (LMNN) metric learning [21, 11] to learn the optimal A from examples of feature vectors representing instances of the same individual's hand (*positives*) and instances of hands of different individuals (*negatives*). A is found by solving the minimization problem

$$\min_{A \succeq 0} \sum_{x,x^+,x^- \in \mathcal{T}} (1-\mu) d_A^2(x,x^+) + \mu \max\{1 + d_A^2(x,x^-) - d_A^2(x,x^+), 0\} \quad (2)$$

where x, x^+, x^- denotes a triplet containing a positive x, x^+ and negative x, x^- pair taken from some training set $\mathcal{T}$ and $\mu \in [0,1]$ is tradeoff parameter. Intuitively, we try to find such A that minimizes the distance between positive pairs x, x^+ (the first term), and at the same tries to achieve a unit margin between the positive and negative distances, $d_A^2(x,x^-) \leq d_A^2(x,x^+) + 1$. The problem is posed and solved as a semidefinite programming problem [21].

We find different metric A_{2D}, A_{3D} for 2D and 3D features, respectively. Since the LMNN metric learning automatically normalizes the feature vectors, the fusion of 2D and 3D feature distances is done by simply taking $\min\{d_{A_{2D}}, d_{A_{3D}}\}$.

3. Results

3.1. Data

To evaluate our system, we collected a proprietary dataset using the Intel RealSense camera. Our dataset consists of approximately 10 instances of hand scans of 100 individuals of ages ranging between 18 and 30 years (in total 1033 instances). The dataset was acquired over the course of six weeks in widely varying environments with different backgrounds and lighting conditions. From our experience, most of the users found the positioning loop intuitive and easy to use.

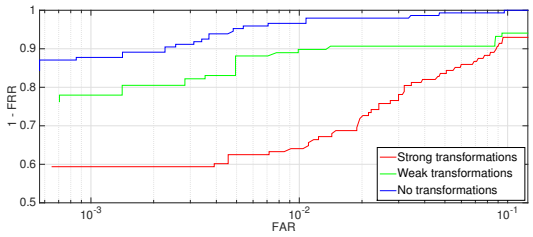


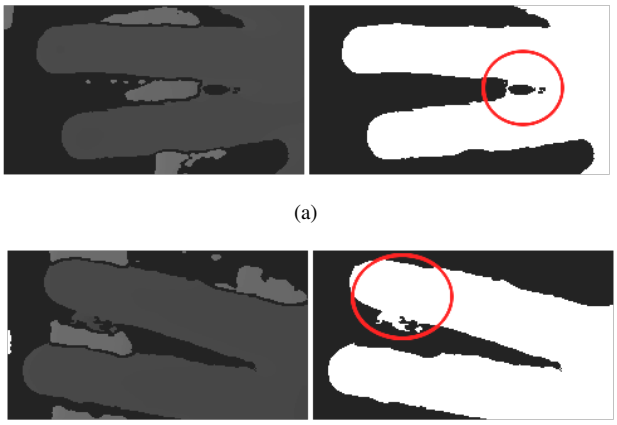
Figure 8. Experiments with hand transformations: Shown are ROC curves for all the cases.

For metric learning, we used a subset of 12 individuals, with approximately 10 instances per user (a total of 124 instances). Testing was performed on the remaining 88 individuals or subsets thereof.

3.2. Evaluation

Positioning loop Our first experiment demonstrates the importance of the positioning phase. For 12 individuals in the dataset, we captured additionally instances of hands under different transformations, such as rotations, fingers attached together, etc. The transformations were categorized into two groups: *weak transformations* on the edge of acceptance of the positioning algorithm, and *strong transformations* that would always be rejected by the positioning algorithm. The performance of the system in the presence of such transformations is captured by the ROC curve in Figure 8. It is clear that without the positioning loop that rules out such problematic cases, the performance of the system degrades significantly. Therefore, the positioning phase is crucial part of the whole system and helps achieving high performance, as shown later in this section.

2D vs 3D features In the second experiment, we compare the robustness of 2D and 3D features on noisy samples. For this purpose, we selected a subset of 173 instances of hands belonging to 17 individuals, which were more difficult to process at the feature extraction stage (typical problems are inaccurate detection of the finger valley or computation of the finger widths, which may occur due to the noise in the input images or hand rotations, that, however, still pass through the positioning phase, shown in Figure 9). The performance of the system on the subset of noisy samples is summarized in Figure 10 and Table 1. It is clearly visible that the aforementioned noise influences mainly 2D features extracted from the hand silhouette. The 3D features are more robust and therefore improve the performance in such problematic cases significantly. Presented results also show that even though we use only very basic 3D features, they are still enough to achieve good performance.



(a) (b)

Figure 9. Examples of wrong feature extraction: (a) Noise causes wrong detection of finger valley; (b) Noise causes wrong computation of finger widths.

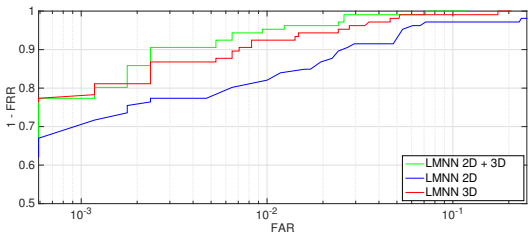


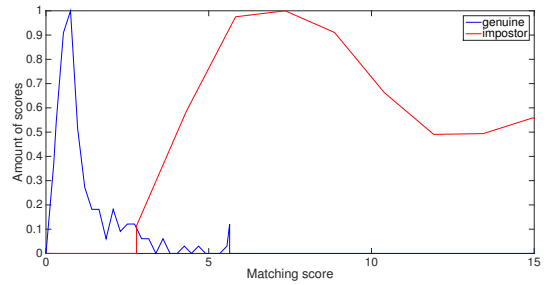
Figure 10. ROC curves of the system evaluated on the subset with the noisy samples demonstrates the importance of the 3D features.

		FRR	
Features	EER	@FAR=0.5%	@FAR=1%
2D	5.36%	22.64%	17.92%
3D	3.77%	13.21%	7.55%
2D + 3D	**2.83%**	**9.43%**	**4.72%**

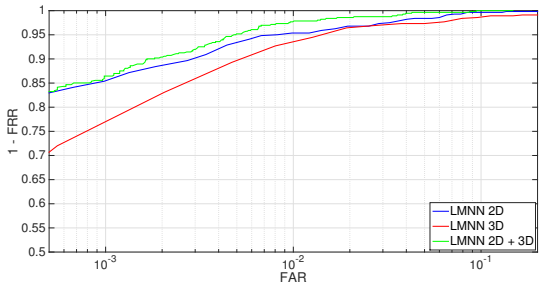
Table 1. Performance of our system evaluated on the subset with the noisy samples. The best performance is achieved from fusion of 2D and 3D features.

Overall performance The last experiments evaluating the overall performance of our system were done using the whole dataset, keeping training and testing samples disjoint. Figure 11(a) shows the genuine and impostor score distributions using a fusion of 2D and 3D features. Figure 11(b) and Table 2 show the performance of recognition based on 2D and 3D features and their combination.

Following Kanhangad *et al*. [9], we report in Table 3 a qualitative comparison of the performance of previous hand biometric recognition methods, showing that our system performs comparably to other approaches. This comparison, however, should be taken with a grain of salt, as different datasets, and therefore also databases of various sizes, were used by different authors. In our case, since the use of a new hardware (Intel RealSense sensor) was a key aspect of the system, we had to resort to a proprietary dataset captured with this sensor.



(a)



(b)

Figure 11. Evaluation of the system: (a) Genuine and impostor score distributions; (b) ROC curve showing comparison of the performance using only 2D features, only 3D features and fusion of both.

		FRR	
Features	EER	@FAR=0.5%	@FAR=1%
2D	2.76%	6.50%	4.63%
3D	2.92%	10.7%	6.42%
2D + 3D	**1.61%**	**4.81%**	**2.23%**

Table 2. Overall performance of our system using LMNN metric learning approach.

4. Conclusions

We presented 3D a biometric hand recognition system based on novel Intel RealSense 3D camera. The input images are acquired in a contactless way, which makes the system more user friendly than many existing commercial hand biometric solutions. As a replacement for "hard" peg-based hand positioning, we use a positioning loop with visual feedback ensuring the quality of captured samples. Our system uses both 2D and 3D features, which are matched using a weighted metric with weights learned from training

Method	Features	Templates	Database	FAR	FRR	EER
Jain and Duta [7]	2D	1 – 14	53	2%	3.5%	N/A
Jain et al. [8]	2D	1 (avg)	50	2%	15%	N/A
Woodard and Flynn [22]	2D + 3D	1 (avg)	177	5.5%	5.5%	5.5%
Malassiotis et al. [16]	2D + 3D	4	73	3.6%	3.6%	3.6%
Kumar et al. [9]	2D + 3D	5	100	5.3%	8.2%	N/A
Kanhangad et al. [10]	2D + 3D	5	177	2.6%	2.6%	2.6%
Our solution	2D + 3D	1 (avg)	88	1.61%	1.61%	1.61%

Table 3. Comparison of our solution to previous hand recognition methods reported in the literature. Note that this comparison is only qualitative, as different datasets were used.

examples by means of LMNN metric learning. Evaluation on a dataset of 88 individuals shows that the combination of 2D and 3D features provides the best performance, and that our performance is comparable with that of state-of-the-art systems based on significantly more expensive sensors. We believe that our results show the potential of using novel cheap 3D sensors that could lead to the development of a new generation of commercial 3D hand recognition products.

Acknowledgements

This research is supported by a CTI grant to the Faculty of Informatics, Universita della Svizzera Italiana, Lugano (17414.1 PFES-ES) and "The IT4Innovations Centre of Excellence" - ED1.1.00/02.0070 (CZ) + internal Brno University of Technology project FIT-S-14-2486.

References

[1] Intel realsense technology - realsense 3d camera, 2014. http://www.intel.com/content/www/us/en/architecture-and-technology/realsense-depth-technologies.html.

[2] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino. 2d and 3d face recognition: A survey. *Pattern Recogn. Lett.*, 28(14):1885–1906, 2007.

[3] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Three-dimensional face recognition. *Int. J. Comput. Vision*, 64(1):5–30, 2005.

[4] K. I. Chang, K. W. Bowyer, and P. J. Flynn. An evaluation of multimodal 2d+3d face biometrics. *IEEE Trans. PAMI*, 4(27):619—-624, 2005.

[5] H. Chen and B. Bhanu. Human ear recognition in 3d. *IEEE Trans. PAMI*, 4(29):718—-737, 2007.

[6] G. Goswami, S. Bharadwaj, M. Vatsa, and R. Singh. On rgb-d face recognition using kinect. *IEEE Conf. BTAS*, pages 1—-6, 2013.

[7] A. K. Jain and N. Duta. Deformable matching of hand shapes for verification. *Proc. ICIP*, pages 857—-861, 1999.

[8] A. K. Jain, A. Ross, and S. Pankanti. A prototype hand geometry-based verification system. *Proc. AVBPA*, pages 166—171, 1999.

[9] V. Kanhangad, A. Kumar, and D. Zhang. Combining 2d and 3d hand geometry features for biometric verification. *Proc. CVPR*, pages 39—-44, 2009.

[10] V. Kanhangad, A. Kumar, and D. Zhang. Contactless and pose invariant biometric identification using hand surface. *IEEE Trans. on Image Processing*, 20(5):1415—-1424, 2011.

[11] B. Kulis. Metric learning: A survey. *Foundations and Trends in Machine Learning*, 5(4):287–364, 2012.

[12] A. Kumar. Incorporating cohort information for reliable palmprint authentication. *Proc. ICVGIP*, pages 583—-590, 2008.

[13] A. Kumar and C. Ravikanth. Personal authentication using finger knuckle surface. *IEEE Trans. Info. Forensics & Security*, 4(1):98—-110, 2009.

[14] A. Kumar, D. C. M. Wong, H. C. Shen, and A. K. Jain. Personal verification using palmprint and hand geometry biometric. *Proc. AVBPA*, pages 668—-678, 2003.

[15] B. Y. L. Li, A. S. Mian, W. Liu, and A. Krishna. Using kinect for face recognition under varying poses, expressions, illumination and disguise. *IEEE Workshop on WACV*, pages 186—-192, 2013.

[16] S. Malassiotis, N. Aifanti, and M. G. Strintzis. Personal authentication using 3-d finger geometry. *IEEE Trans. Info. Forensics & Security*, 1(1):12—-21, 2006.

[17] S. Ribaric and I. Fratric. A biometric identification system based on eigenpalm and eigenfinger features. *IEEE Trans. PAMI*, 27(11):1698—-1709, 2005.

[18] G. Shakhnarovich. Learning task-specific similarity, 2005. PhD thesis, MIT.

[19] M. Sonka, V. Hlavac, and R. Boyle. Image processing, analysis, and machine vision, 1999. PWS Publishing.

[20] S. Suzuki and K. Abe. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1):32—46, 1985.

[21] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

[22] D. L. Woodard and P. J. Flynn. Finger surface as a biometric identifier. *CVIU*, 100(3):357—-384, 2005.

[23] W. Xiong, K. A. Toh, W. Y. Yau, and X. Jiang. Model-guided deformable hand shape recognition without positioning aids. *Pattern Recognition*, 38(10):1651—-1664, 2005.

[24] D. Zhang and G. Lu. *3D Biometrics: Systems and Applications*, volume 1, pages 85 – 170. Springer-Verlag New York, 2013.