2015 IEEE International Conference on Systems, Man, and Cybernetics

An Automatic Two-hand 2D:4D Finger-ratio Measurement Algorithm for Flatbed Scanned Images

Frode Eika Sandnes[1,2]
[1]Faculty of Technology Art and Design, Oslo and Akershus University College of Applied Sciences, Oslo, Norway
[2]Faculty of Technology, Westerdals school of Art, Communication and Technology, Oslo, Norway
Frode-Eika.Sandnes@hioa.no

***Abstract*— There is a vast body of research based on 2D:4D finger ratio measurements within the fields of medicine, physiology, and several other disciplines. There are well-known conventions for how such finger measures should be conducted, and all such measurements are to this date manual. Manual measurements are time consuming and prone to human error, different practices and subjective influences. This study proposes a novel algorithm for automatically computing the 2D:4D finger ratios using images of hand pairs. The algorithm is simple to implement and robust to image noise. Initial experimental assessments indicate that the algorithm successfully manages to accurately determine 2D:4D finger ratios. The proposed algorithm could help improve the research fields that make use of digit ratio measurements. Areas of algorithm improvements are also identified.**

Keywords- 2D:4D finger ratio, image analysis, measurements

I. Introduction

Digit ratio research has received significant amount of research attention in recent decades [1, 2, 3]. The digit ratio is believed to serve as an approximate measure for prenatal androgen exposure. The advantage of the digit ratio is that it is easy to understand and relatively easy to measure. It is measured by dividing the length of the index finger by the length of the ring finger. The length of a finger is the line segment going from the tip of the finger to the crevice where the finger is attached to the hand. To obtain more accurate reading it is recommended to measure the digit ratio of both hands and use the average of the two measurements.

Digit ratio measurements have until now been performed manually. To the best of the author's knowledge there is only one previously published study on an automatic method for measuring digit ratios [4]. The method proposed in [4] is a rotation invariant approach intended for one hand with constraints on minimum intensity differences between the hand and the background. The method was proposed to be used with a mobile phone where the hand was overexposed by the camera flash. Other studies involving automatic observation of fingers and hands are in the domain of gesture recognition [5, 6, 7].

This study proposes a novel method for detecting finger ratios given two hands with more relaxed constraints on hand background differences. The method is not rotation invariant and makes certain assumptions about the input image. It is intended as a more accurate measurement tool for traditional finger ratio researchers.

II. Method

The novel 2D:4D ratio extraction algorithm presented starts by separating the hands from the background using a domain specific and robust image binarization algorithm. Next, the outline of the hand is traced and estimates of hand feature points are determined. These hand feature points are fine tunes and used as basis for determining the finger vectors. Finally, the detected finger vectors are used to calculate the digit finger ratios for the two hands. The details of each of these steps are outlined in the subsequent sections.

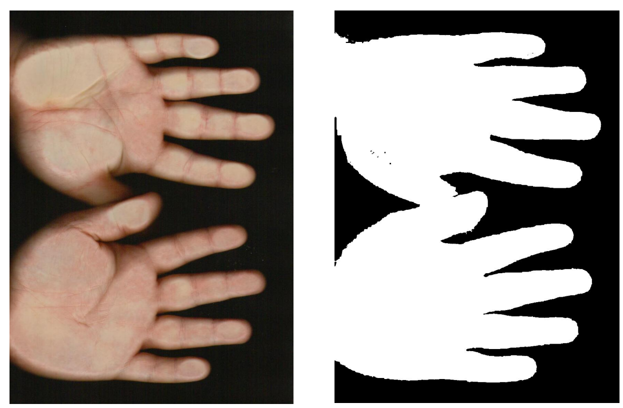

Figure 1. A) An example flatbed scan of two hands facing down on the glass. b) Binarized image.

A. Assumptions

The algorithm proposed herein is based on a number of assumptions. First, the input images must be in color. This assumption is essential for the hands to be successfully separated from the background.

Second, it is assumed that the images are obtained preferably using an A3 flatbed scanner where both of the hands are placed face down on the glass for scanning. With a known scan format one can also extract absolute length measurements. However, since 2D:4D measurements are ratios they do not depend on the absolute measurement unit used as the

978-1-4799-8697-2/15 $31.00 © 2015 IEEE
DOI 10.1109/SMC.2015.215

computations performed by the algorithms is based on image pixels which is a relative unit of measure.

The scanned images are assumed to be in portrait orientation with the left hand on the upper part of the image and the right hand on the bottom part of the image. The two thumbs are thus pointing towards each other in the middle. The arms come from the left side with the fingertips pointing in the right direction (see Fig. 1). This configuration is the result of the operator typically standing on one side of the flatbed scanner while taking measurements with the participant standing on the opposite side of the scanner. It is also assumed that the scanner cover is closed on top of the hands ensuring that the background is sufficiently unsaturated and thus different from the color of the skin of the hands.

We further assume that the hands scanned contain all the ten fingers and that any jewelry is removed. Finally, it is assumed that the fingers are spread out such that the background shines through between the fingers, that is, the fingers must not stick together.

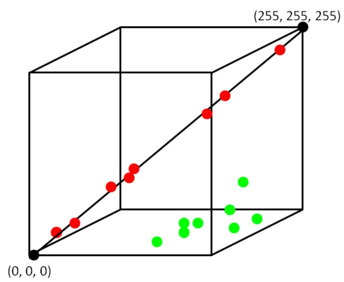

Figure 2. Separating hand and background using the background diagonal and hand points in the RGB colour space.

B. Hand and background separation: Image binarization

The first step of the algorithm is to separate the hand from the background. This is achieved by binarizing the image. Most of the basic threshold based binarization algorithms do not fulfill the task since the background may vary from white to black via shades of grey due to the shadows cast by the fingers as the scanner light source passes the hand.

The background can thus be described as all the completely unsaturated points in the RGB color space that follows the color cube diagonal from black via shades of grey to white. This diagonal is represented by the line:

$$\frac{1}{2}r+\frac{1}{2}g-b=0 \tag{1}$$

Here the r, g and b components are in the range 0 to 255. Alternatively this can be represented as

$$f(x)=Rx+Gx+Bx \tag{2}$$

Where R, G and B are unit vectors in the RGB color space, that is, [1, 0, 0], [0, 1, 0] and [0, 0, 1], respectively. The hand in the image has a visible saturated hue. The color points associated with the hand will therefore not fall onto this diagonal. A given point c in the color space is therefore classified as part of the background if it is sufficiently close to the diagonal, that is

$$\left|f(x)-c\right|<T \tag{3}$$

Otherwise, c is part of the hand (see Fig. 2). The threshold T was set to 20 through experimentation and a Manhattan distance function was used due to its simple computation. Fig. 2 shows binarization using this approach.

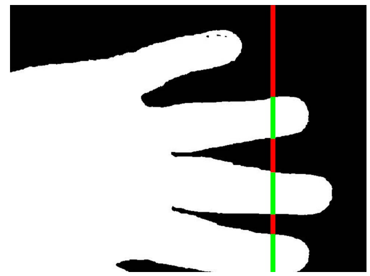

Figure 3. Vertical scanning for fingers.

C. Hand outline extraction

The outline of the hand is obtained from the binarized image in three steps. First, the image is scanned from right to left with one vertical line at a time from top to bottom (starting at y=0). That is, the image is scanned in the direction from the end of the image towards the fingertips. The scan is stopped at some point towards the left as there is no need to detect the outline of the base of the hand and arm. Images can potentially also contain participant labels in this region. Experimentation revealed that a suitable stop point is at 0.3 of the image width, that is at $x = 0.3\cdot\max$.

Once a pixel change is detected a potential finger candidate is found and the start point y_{start} is recorded. That is if

$$p(x,y_i)\neq p(x,y_{i-1}) \tag{4}$$

Where p is the pixel value at x, y. The end of the finger candidate is detected once the pixel changes back to the background color and this point y_{end} is recorded.

Next, the a check is performed to see if the finger candidate segment $[y_{start}, y_{end}]$ is the result of noise or the detection of a real finger. This is achieved by determining if the length of the segment is beyond a minimum threshold W, namely

$$y_{end}-y_{start}>W \tag{5}$$

The start and end points of segments passing this test are stored as trace points in a list M_x, otherwise they are discarded. The threshold W was set to

$$W=\frac{y_{\max}}{100} \tag{6}$$

Initially, the number of segments resulting from the vertical scan will be zero, that is

$$\left|M_x\right|=0 \tag{7}$$

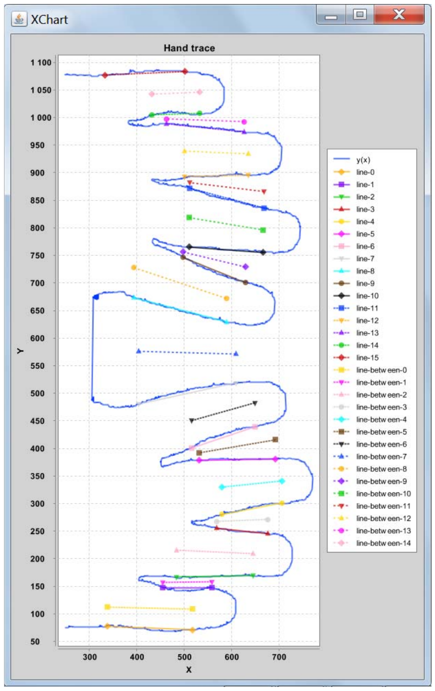

Figure 4. Hand outline with finger outline segments and finger mid-lines.

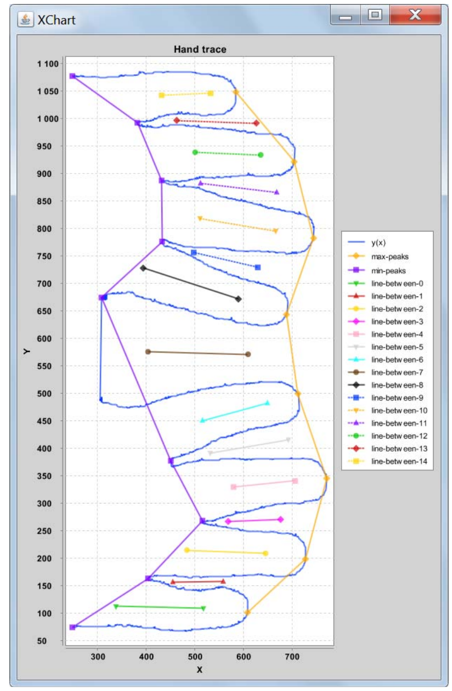

Figure 5. Crevice points, fingertips and finger vectors.

But, once the tip of the first finger is detected the number of segments increases to 2. As we scan downwards the x-axis the number of segments will increase to 16, which means that all the fingers beside the thumb have been detected. Next, the number of segments will decrease as the scan reaches the crevices of the fingers, and of course increase once the thumbs are detected. In other words, an interesting finger feature point is encountered once there is a change in the number of segments resulting from a scan, namely

$$|M_x| \neq |M_{x+1}| \tag{8}$$

The vertical scan process is illustrated in Fig. 3. The second part of the hand outline extraction involves combining the traces stored in *M*. The algorithm starts at the point x where there are four fingers at each and detected, that is

$$|M_x| = 16 \tag{9}$$

The scan points in M_x are ordered according to increasing *y*-value and hence also according to correct finger order, that is from right little finger, ring finger, etc up to left ring finger and left little finger since the hands are mirrored. These points are therefore used as starting points for the eight outline segments in *U*.

First, *M* is processed from this *x* value to x_{max}. Then, *M* is processed in the opposite direction from *x* value down to the stop point on the x-axis. For each processing step for a given *x* the points in M_x are added to the closest outline segment *U*.

However, if a point in M_x is further away in *y*-value than a threshold *W* the point is discarded. Moreover, if the distance between the point in M_x amd the closest outline point in the *x*-direction is more than 1 the point is also discarded.

In other words, the point $M_{x,i}$ is assigned $U_{x,j}$ if

$$\min_j |M_{x,j} - U_{x-1,j}| \tag{10}$$

And

$$|M_{x,j} - U_{x-1,j}| < W \tag{11}$$

These checks prevent the thumb outlines from being recorded.

The final stage of the finger outline extraction process is to combine the outline segments in *U* into one trace *S* and determine the key hand feature points, namely crevices *C* and fingertips *D*. This is achieved by traversing the segments in *U* in alternating directions starting from the first going to the last segment. The first segment is scanned from the smallest *x*-value going up to the maximum *x*-value, the second segment is traversed going from its maximum *x*-value to its minimum value, etc. The first point of the trace is labeled the first crevice contender, and the end of the first segment is labeled the first fingertip contender. For each segment alternating finger crevice and fingertip contenders are recorded. Fig. 4 shows the outline of the hand in Fig. 1.

D. Finger vector detection

The next step of the algorithm is to determine the feature points constituting the line segments that are used to compute the 2D:4D finger ratios. This involves fine tuning the fingertip points *D* and determining the roots *R* of each finger. This is because the initial estimate may be somewhat inaccurate as it is the result of the tangent with the vertical scan line and not according to a tangent perpendicular to the direction of each finger. To fine tune the fingertip and crevice points, lines passing through the middle of each finger are found.

To find this finger midline the line for the top and bottom edge of each finger is found. A robust approach devised to use the first quartile point and the third quartile point on the curve going from the crevice point and fingertip point defining the curve segment as basis for the midline. The first quartile point is located at position

$$a = \frac{3}{4}S(F_i) + \frac{1}{4}S(F_{i+1}) \tag{12}$$

Here F is an ordered list of crevice and fingertip index points along the hand outline curve S. Similarly, the third quartile point is

$$b = \frac{1}{4}S(F_i) + \frac{3}{4}S(F_{i+1}) \tag{13}$$

The respective line segment

$$y = Ax + B \tag{14}$$

Is thus given by

$$A = \frac{b_y - a_y}{b_x - a_x} \text{ and } B = a_y - Aa_x \tag{15}$$

Fig. 4 shows the finger contour lines for the hands in Fig. 1. The line $f(x)=Ax+B$ between two lines $y=A_1x+B_1$ and $y=A_2x+B_2$ is thus found by finding their intersection, namely

$$x_c = \frac{B_2 - B_1}{A_1 - A_2} \tag{16}$$

The slope of the midline is the mean of the two respective slopes, namely

$$A = \frac{A_1 + A_2}{2} \tag{17}$$

And B is thus simply

$$B = A_1 x_c + B_1 - Ax_c \tag{18}$$

Fig. 4 shows the contour lines for the finger midlines for the hands in Fig. 1. Each crevice and fingertip points are thus updated according to where midlines cross the hand outline curve S, namely

$$\min_x \left| f_{mid}(x) - S_x \right| \tag{19}$$

The process of fine tuning the crevice and fingertip points is illustrated in Fig. 5.

The root R of the ring fingers are simply found as the midpoint of the two crevice points at the left and right side of the finger, or more specifically for finger i.

$$R_i = \frac{C_i + C_{i+1}}{2} \tag{20}$$

Here C_i, C_{i+1} represent the two crevice points and R_i the root point of the finger. Fig. 6 illustrates the process of estimating the root of the fingers.

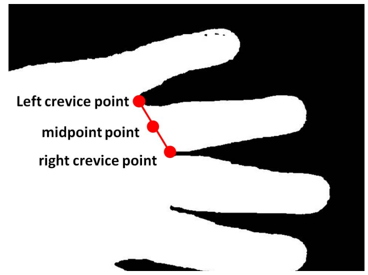

Figure 6. Estimating finger roots for the ring fingers.

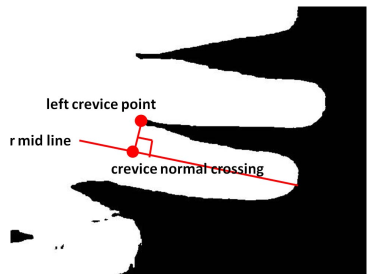

Figure 7. Estimating finger roots for the index fingers.

The root point for the index finger has only one reliable crevice point. Instead, the estimate of the root is just found as the point where the normal of the midline $y=Ax+B$ of the index finger that goes through the crevice point $C=(C_x,X_y)$ cross, namely

$$R_x = \frac{C_x + AC_y - AB}{A^2 + 1} \text{ and } R_y = AR_x + B \tag{21}$$

Here, R_x computed in the first expression is used as part of the second expression. Fig. 7 illustrates the process of computing the root point based on the normal to the finger midline.

E. Digit ratio computation

Once reliable finger root points R and fingertip points D have been determined, the finger ratio for finger i and j is simply the ratio of the two finger lengths defined by the vector going from the finger root R to the fingertip D (see Fig. 5):

$$DR(i,j) = \frac{|R_i - D_i|}{|R_j - D_j|} \tag{22}$$

The digit ratio for both hands are computed as

$$DR = \frac{DR(2,4) + DR(5,7)}{2} \tag{23}$$

Here it is assumed that the fingers are numbered in order from 1 to 8 and the two thumbs are not counted.

Figure 8. A3 scan closest to the mean.

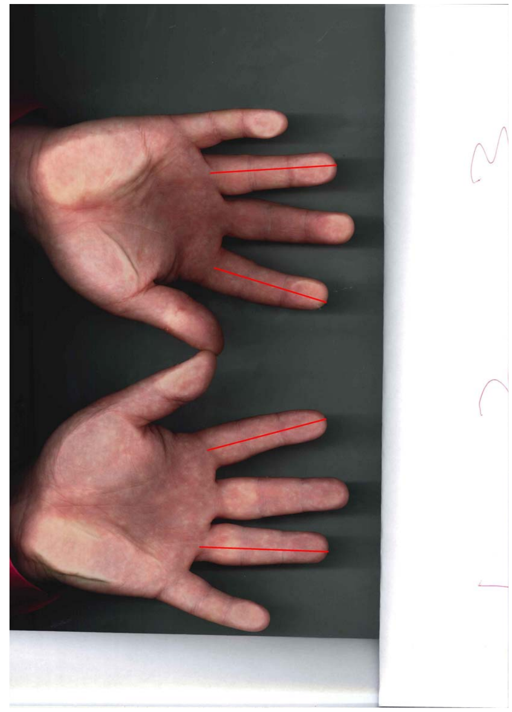

Figure 9. A3 scan most different to the mean.

III. EXPERIMENTS

Some basic testing is presented in this initial study to demonstrate the effectiveness of the algorithm. The tests are based on the authors´ own hands and the correctness of the results is based on visual inspection and the consistency of the measurements. Eight images were acquired using a Ricoh office standard A3 multifunction flatbed scanner and six hand images were acquired using an OKI MP2500 A4 multifunction flatbed scanner. The resolution of the A3 images were 3508 x 4961 pixels (300 dpi) and the resolution of the A4 images where 827 x 1146 pixels (100 dpi). The images acquired with the A3 scanner also include some pieces of papers that were placed to trigger the sensors activating the A3 scanner mode. However, the results demonstrate that the segmentation algorithm is robust to these pieces of paper as they do not interfere with the results. Figs. 8 and 9 show the A3-scanned image with superimposed measurement points that are closest to the measured mean and the one that is the most far away from the mean, respectively. Figs. 10, 11 and 12 show three images obtained with the A3 scanner with automatically superimposed measurement points.

The results achieved with the eight A3 scans are provided in Table 1. The results show that the overall standard deviation for both hands is 0.01 which represents about 1% error. Fig. 9 (plot 4) which shows the hand with the largest deviation from the mean reveals that the resulting measurement is too far off. In conclusion, the algorithm successfully manages to extract the finger digit ratios from the A3 scanned images with acceptable consistency.

The A4 scans were carried out with three different scanning scenarios, namely (a) using a semi transparent film on top of the glass with the cover closed on top of the hands, (b) without semi-transparent film with the cover closed on top of the hands and finally (c) without semi-transparent film with the cover open. The purpose of the semi-transparent film is to hide the fingerprints and to make the background more blurred.

The scans obtained with the A4 scanner were less accurate. This is mostly because there were not enough space to spread out the fingers on the glass plate of the scanner and the algorithm therefore was not able to separate fingers that were too close together. The algorithm is therefore only capable of successfully finding the correct finger ratio for one of the hands.

The algorithm was able to extract correct information from the parts of the image that adhered to the constraints. That is, these scans produced acceptable measurement for at least one hand for each of the six conditions. The digit measurement results are shown in Table 1. Note that the D2 and 4D length measurement units are listed in pixels.

Still, the results show that in order to make the algorithm more robust more attention needs to be paid to detecting crevices more accurately in situations where fingers are too close together. This will be the focus of the next phase of this research.

The algorithm was also measured in terms of timing to assess the computational effectiveness of the approach. Timing measurements reveals that the mean time to process the larger A3 images was 33.4 seconds (standard deviation (SD) = 4.0) and the mean time to process the smaller A4 images was 6.8

seconds (SD = 8.6) where the most time was taken processing the first image. Clearly, the time to process the images are in the range of seconds and thus it is feasible to apply the method to large amount of input data.

TABLE I. DIGIT RATIO MEASUREMENTS (A3 SCANS, PIXELS)

	overall	left			right			diff
No.	ratio	ratio	D2	D4	ratio	D2	D4	left right
1	0,96	0,96	867	906	0,95	859	900	0,003
2	0,97	0,96	864	897	0,98	880	895	-0,020
3	0,96	0,95	850	897	0,97	869	893	-0,026
4	0,94	0,95	854	899	0,93	818	882	0,023
5	0,97	0,98	874	895	0,95	855	897	0,023
6	0,97	0,97	881	907	0,97	887	913	0,000
7	0,97	0,97	878	906	0,98	889	910	-0,009
8	0,96	0,96	867	902	0,97	881	911	-0,007
Mean	0,96	0,96	866,92	901,25	0,96	867,33	900,10	0,00
SD	0,01	0,01	10,90	4,88	0,02	23,42	10,72	0,02

IV. CONCLUSIONS

This paper has, to the best of the author's knowledge, presented the first ever automatic 2D:4D finger ratio measurement algorithm based on image scans of two hands. The initial results indicate that the algorithm performs comparably to manual measurements. The algorithm appears robust to noise in the input image in terms of backgrounds. The algorithm is simple to implement as the current implementation comprises approximately 500 lines of java code. The algorithm has acceptable performance considering that many simple optimizations are possible. However, performance is not a key issue for finger ratio measurements as it probably suffices to make offline processing.

Future work involves more extensive testing on larger datasets including participants with different skin colors and hand shapes. Improvements include making the algorithm robust to fingers sticking together, invariant to rotation and more accurate finger root detection using finger characteristics such as finger widths skin shadow and texture features. A more robust method for detecting crevices where fingers are close together is needed.

ACKNOWLEDGEMENTS

The author is thankful to Levent Neyse for valuable input and to Yo-Ping Huang for valuable encouragements.

REFERENCES

[1] D. A. Putza, S. J. C. Gaulinb, R. J. Sporterc and D. H. McBurney, "Sex hormones and finger length What does 2D:4D indicate?" Evolution and Human Behavior 25: 182–199, 2004.

[2] M. Voracek, J. Pietschnig, I. W. Nader and S. Stieger, "Digit ratio (2D:4D) and sex-role orientation: Further evidence and meta-analysis," Personality and Individual Differences 51: 417–422, 2011.

[3] M. H. McIntyre, E. S. Barrett, R. McDermott. D. D. P. Johnson, J. Cowden and S. P. Rosen, "Finger length ratio (2D:4D) and sex differences in aggression during a simulated war game," Personality and Individual Differences 42: 755–764, 2007.

[4] F. E. Sandnes, "Measuring 2D: 4D finger length ratios with Smartphone Cameras". In IEEE International Conference on Systems, Man and Cybernetics (SMC), IEEE, pp. 1697-1701, 2014.

[5] V. I. Pavlovic, R. Sharma and T. S. Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review," IEEE Transactions on Pattern Analysis and machine Intelligence 19 (7), 1997.

[6] W. T. Freeman and M. Roth, "Orientation Histograms for Hand Gesture Recognition," Technical report, Mitsubishi Electric Research Laboratories, Cambridge Research Center, TR-94-03a, 1994.

[7] M. Fukumoto, Y. Suenaga and K. Mase, ""Finger-Pointer": Pointing interface by image processing," Comput. & Graphics 18(5): 633-642, 1994.

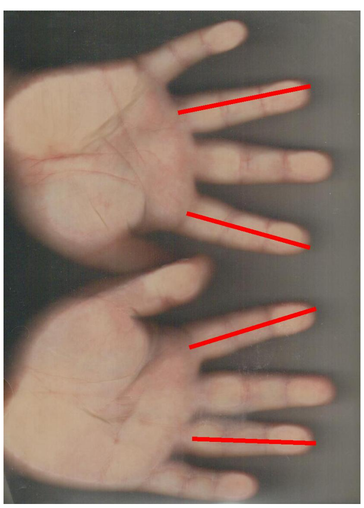

Figure 10. Results with semi-transparent film on top of the glass.

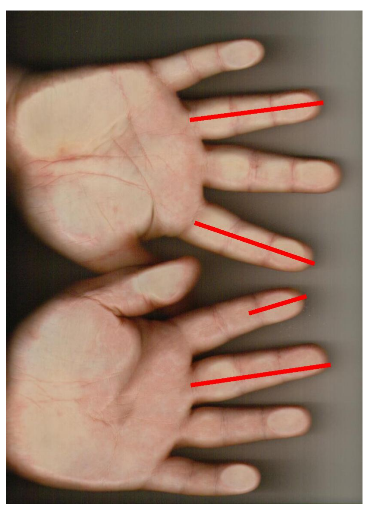

Figure 11. Results with the cover closed on top of the hands.

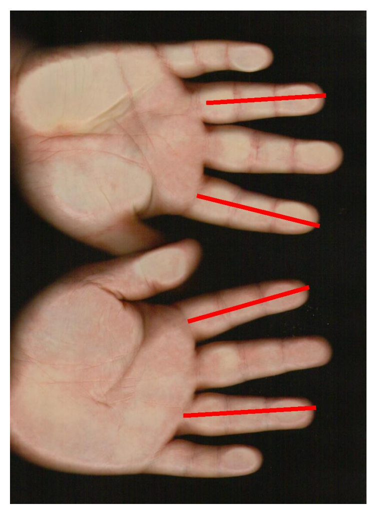

Figure 12. Results with the cover open.