

Determining the Bias and Variance of a Deterministic Finger-Tracking Algorithm

Valerie S. Morash

Department of Psychology, University of California, Berkeley

Bas H.M. van der Velden

Image Sciences Institute, University Medical Center Utrecht

Abstract

Finger tracking has the potential to expand haptic research and applications, as eye tracking has done in vision research. In research applications, it is desirable to know the bias and variance associated with a finger-tracking method. However, assessing the bias and variance of a deterministic method is not straightforward. Multiple measurements of the same finger position data will not produce different results, implying zero variance. Here, we present a method of assessing deterministic finger-tracking variance and bias through comparison to a non-deterministic measure. A proof-of-concept is presented using a video-based finger-tracking algorithm, developed for the specific purpose of tracking participant fingers during a psychological research study. The algorithm uses ridge detection on videos of the participant's hand, and estimates the location of the right index fingertip. The algorithm was evaluated using data from four participants, who explored tactile maps using only their right index finger and all right-hand fingers. The algorithm identified the index fingertip in 99.78% of one-finger video frames, and 97.55% of five-finger video frames. Although the algorithm produced slightly biased and more dispersed estimates relative to a human coder, these differences ($x = 0.08$ cm, $y = 0.04$ cm) and standard deviations ($\sigma_x = 0.16$ cm, $\sigma_y = 0.21$ cm) were small compared to the size of a fingertip (1.5 - 2.0 cm). Some example finger-tracking results are provided, where corrections are made using the bias and variance estimates.

Keywords: Haptics, finger tracking, bias, variance, algorithm

Determining the Bias and Variance of a Deterministic Finger-Tracking Algorithm

Introduction

Finger tracking is analogous to eye tracking, which is readily used for visual-perception research (Hayhoe & Ballard, 2005; Yarbus, 1967), medical diagnosis (Green et al., 2009), and human-computer interaction (Jacob & Karn, 2003). However, there has been less emphasis on finger tracking than eye tracking, although it could find similar applications. This is particularly true in psychology, where eye tracking was first used in the late 19th and early 20th centuries to suggest the cognitive processes that support reading (results first reported in Huey, 1908). In contrast, finger tracking has only gained attention in recent decades. This may be due to the greater complexity associated with tracking relatively far-moving fingers compared to determining the orientation of a stationary eyeball. Alternatively, vision has long been the dominant sense of study in psychology, naturally resulting in fewer devices and less research on finger tracking than eye tracking.

A notable exception is in the field of human-computer interaction, where the objective is to allow humans to interact with computers through naturalistic hand gestures. Gesture analysis typically consists of two sequential phases: first, relevant image features are extracted and, second, these image features are used to compute gesture-recognition model parameters (Pavlovic, Sharma, & Huang, 1997). It is within the first stage of image-feature extraction that a gesture recognition system may involve fingertip detection and tracking, the methods of which may be extended to finger tracking for psychological research. Fingertip detection and tracking is accomplished through either invasive or noninvasive means, similar to eye tracking devices, which track gaze using either non-invasive optical methods or invasive contact lens and eye-coil approaches (Kimmel, Mammo, & Newsome, 2012). Invasive finger tracking methods ask the participant to wear electromagnetic or colored markers on the hand or fingers, which greatly simplifies signal/image processing through approaches such as hue segmentation (Kim & Fellner,

2004; Pavlovic et al., 1997; Wang & Popović, 2009). However, gloves or thimbles that cover the fingers will impeded haptic perception (Klatzky, Loomis, Lederman, Wake, & Fujita, 1993), making these approaches undesirable in perceptual research.

Non-invasive optical systems acquire and process video images of the hand. Fingertip positions can be detected using template matching, which examines the correlation between sections of a video image and a fingertip template, a detailed or simplified picture of a fingertip or finger (Zabulis, Baltzakis, & Argyros, 2009). Alternatively, characteristic features of the finger may be used for detection. For example, when reading braille on a clear sheet, the finger may be pressed against the smooth braille sheet with enough pressure that it appears white when recorded from below (Breidegard, 2007); or the changing curvature of a finger's outline, from low to high to low, can be used as a heuristic for identifying fingertips (Maggioni, 1995). This latter method is closest to our approach, which uses the fact that fingertips are associated with high “ridge” values.

For finger tracking in the context of psychological research, it is important to establish the finger-tracking method's bias and variance. Establishing that the bias of a finger-tracking method is low or near zero ensures that the method produces valid estimates that correspond well with “true” fingertip positions. Low variance is also desirable, as it indicates that real changes in finger position will be apparent in the estimated finger positions, and not obscured by random estimation error. The bias and variance of the finger tracking method can be used in interpreting finger-tracking results, either directly by making numerical corrections to results, e.g., by correcting location to offset a known bias, or in tempering the scientist's interpretation. The precision of two finger-tracking methods may also be compared directly based on their bias and variance.

Gesture recognition systems do not report the bias and variance of finger position estimation, which is an intermediate result in the algorithm, and are instead appropriately evaluated based on the accuracy of the final outputted gestures. Finger tracking in psychological research often falls back to reporting the variance associated with the

tracking apparatus or sensors (Farinella & Rustico, 2008; van Polanen, Tiest, & Kappers, 2011). However, the variance of the sensor can only be considered a lower-bound of the fingertip estimation variance. For example, in a scheme that tracks a marker that is taped to the fingernail, the marker's position is a poor estimate of the fingertip's lateral position when the finger curls or rotates. In such an event, the marker is no longer above the fingertip, illustrating that the variance and bias of the fingertip estimate is not strictly that of the sensor's ability to track the marker when not on a finger.

Rather than report measurements of the sensor, some finger-tracking reports simply do not address the characteristics of their fingertip estimates' bias and variance (e.g., Breidegard, 2007). This is not unreasonable, considering that it is not obvious how to establish the variance of a deterministic finger-tracking method. The straightforward approach to measuring variance would be to run the method/algorithm on the same finger position data multiple times. However, being deterministic, there will be no variance in the estimates across multiple passes on the same data. In the current report, we overcome this issue by comparing the results of a deterministic algorithm to that of a non-deterministic one with calculable variance. Our algorithm tracks finger position over a two-dimensional tactile stimulus. This is analogous to the psychological study of eye movements over two-dimensional visual stimuli presented on a computer screen. Measurements were compared to those from a human coder, but any non-deterministic method could be used for comparison.

The main contributions of this report include the following. First, we describe the development of a finger-tracking algorithm using ridge detection, tailored to finger tracking in psychological research. This algorithm is easily implemented using low-cost materials, and is thoroughly described so that other researchers may find it easy to implement. Second, we measure the bias and variance of this algorithm. Our approach is to compare results from the deterministic algorithm to that with a known bias and variance, either known a priori or measurable if the approach is non-deterministic. Third, we provide

several psychological research applications, with preliminary results, to illustrate the use of finger tracking with measured bias and variance.

Methods

Data Collection

Our finger-tracking algorithm using ridge detection was evaluated by applying it to videos collected during a psychological study. The data were from four right-handed sighted participants (two female, ages 26, 32, 35, and 38). Each participant provided informed consent prior to enrolling in the study, and the protocol was approved by University of California, Berkeley's Committee for Protection of Human Subjects.

The stimuli we used for developing our finger-tracking algorithm were selected from the Tactile-Map Open Stimulus Set (TMOSS; Morash, Connell Pensky, & Miele, 2012a, 2012b). These stimuli are tactile maps, which are similar to line-based visual maps, but features have been raised and/or textured to make them accessible to touch. These maps are designed to the same specifications as tactile maps for users who are blind. Also, these stimuli have been tightly controlled to make them useable in randomized psychological experiments (Morash, Connell Pensky, & Miele, 2013). The tactile maps were laser-cut from clear acrylic. Each map was 30.5 cm x 30.5 cm in size and represented a fictitious park, containing an inset rough-textured area to represent a lake, raised lines to represent walking paths, and point symbols (raised-line squares, circles, triangles, Ts, and ovals) to represent locations of restrooms, trash cans, etc. An example stimulus can be seen later in the results, Figure 7.

Procedure. Each participant was blindfolded and asked to search for a landmark on 16 tactile maps. Landmarks were clusters of symbols on a map, such as a cluster of square symbols. The participants were trained how to recognize map symbols and clusters before beginning the study. On 8 of the maps, the participants were allowed to use all five fingers on their right hand. On the other 8 maps, participants were allowed to use only

their right index finger, and the other fingers were taped to the palm. Collecting data from both one- and five-finger conditions, which was done in alternating order, allowed us to assess the performance of our algorithm for extracting index fingertip position in both one- and five-finger applications. Although using all five fingers may appear to be the most natural for haptic exploration, many participants in haptics studies will spontaneously use only their index finger (Symmons & Richardson, 2000).

During a trial, a map was placed in front of the participant on a clear table. The participant's index finger was placed in approximately the center of the map. The pace of the study was controlled by a computer, which played auditory instructions for the participant to search for a landmark, e.g., "Please locate the cluster of squares and say 'here.'" The computer then played a beep, after which the participant could begin searching the map for the landmark. Once the participant had found the landmark, he or she said "here," and removed his or her hand.

As mentioned, the stimuli we used for developing our finger-tracking algorithm were made from clear acrylic, which provided the opportunity to video record hand movements from below the stimuli, similar to previous works on finger tracking in psychology (Breidegard, 2007; Millar, 1988). Below the clear table and tactile map a video camera was positioned (Canon Vixia HF R21, with a Canon 3.28-megapixel full HD CMOS image sensor). It pointed upwards and was focused to the depth plane of the tactile map. The camera's field of view was filled by the tactile map (30.5 cm x 30.5 cm) plus some additional space on each side (roughly 8 cm). The video camera recorded at a 1920 x 1080 resolution at 29.97 frames per second (30 Hz). Above the table was a white shelf that provided the video a white backdrop. The setup was the same as in Morash et al. (2013), which contains a detailed drawing of the setup, the only difference being a different camera model. Exploring the tactile map stimuli did not consistently result in the participants' fingertips "turning white" (as in Breidegard, 2007), possibly due to visual obstruction by the raised and textured elements on the maps or the participants did not press down on

the stimuli hard enough for their fingertips to appear white.

Algorithm

The first step in our algorithm is to convert each frame from red-green-blue (RGB) values to luminance values (i.e., grayscale images). Luminance values from a background image with no hand are then subtracted from each frame's luminance values. This removes many of the stimulus features from the frame. The subtracted image is then left-right flipped to compensate for recording from below.

Ridge detection is applied to the subtracted and left-right flipped image values. Ridge detection is a rotationally invariant detector of light-dark-light transitions (maximally positive), and dark-light-dark transitions (maximally negative). Therefore, it does not matter how the light-dark-light portion of the image is oriented. This involves computing the intrinsic (w, v) coordinate system, which is aligned to the local image gradient. The w axis is defined parallel to the luminance gradient, in the direction of maximum luminance change. The v axis is defined as perpendicular to the w axis, in the direction of minimum luminance change. These coordinates are referred to as gauge coordinates (ter Haar Romeny, 2003, pp. 91-136). The change in coordinate system is illustrated in Figure 1.

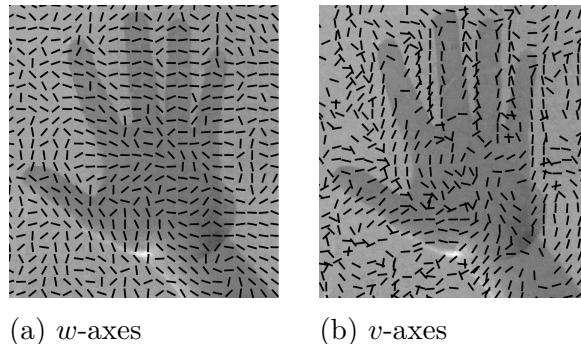


Figure 1. Intrinsic coordinate system (w, v) defined at image points (x_i, y_i) .

Note that in Figure 1a, the locally defined w axes lay perpendicular to the contours of the hand, while the v axes are orthogonal to the w axes, and generally follow the

contours of the hand. This is particularly apparent along the right edge of the hand image.

Conversion from the extrinsic (x, y) coordinate system to the intrinsic (w, v) coordinate system is defined by rotating the local (x, y) system at point (x_i, y_i) by angle θ ,

$$\begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix} = \frac{1}{(L_x^2 + L_y^2)} \begin{pmatrix} L_x \\ L_y \end{pmatrix} \Big|_{(x_i, y_i)} \quad (1)$$

L is used, generally, to denote the convolution of a Gaussian kernel g with the image f , $L = g * f$ (Lindeberg, 1998). Subscripts of L and g denote the axes and order of differentiation, i.e., $L_{x^\alpha y^\beta}$ indicates the image convolved with a Gaussian kernel $g_{x^\alpha y^\beta}$ (Lindeberg, 1998).

Convolving the subtracted image with a second-order Gaussian kernel g_{vv} reveals the “ridges” (Lindeberg, 1998). The scale of the ridge detector $L_{vv} = g_{vv} * f$ is defined by the standard deviation (σ) of the Gaussian kernel. Different scales extract different image features. In other words, if one looks for bumps about 1 to 0.5 cm sized, the fingertip shows up (Figs. 2a & 2d). A larger bump is a finger (Figs. 2b & 2e). A still larger bump is the hand itself (Figs. 2c & 2f).

The shape of a second-order Gaussian kernel is shown in Figure 3. Convolving this kernel with an image produces a maximum value when the image transitions from light to dark to light, along the local v axis, on the same scale as the kernel. With the right scale ($\sigma = 0.5$ cm), this maximum occurs on the fingertips.

In our implementation, transformation to gauge coordinates and convolution with a second-order Gaussian kernel were done with a single operation (ter Haar Romeny, 2013),

$$L_{vv} = \frac{L_x^2 L_{yy} - 2L_x L_y L_{xy} + L_y^2 L_{xx}}{L_x^2 + L_y^2} \quad (2)$$

Therefore, although ridge detection can be conceptualized as two steps, transformation to gauge coordinates through calculation of θ , followed by convolution with a second-order

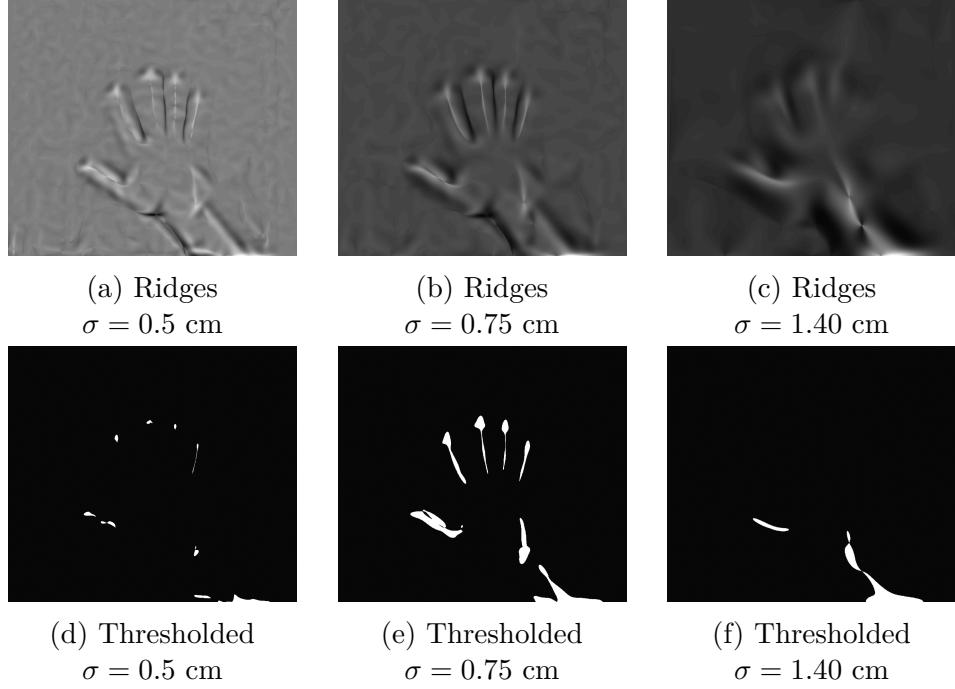


Figure 2. Ridges and thresholded ridges, different σ s.

Gaussian kernel, we accomplished ridge detection with a single operation defined in Equation 2.

Following computation of the ridge detector (L_{vv}), the image was thresholded. This revealed clusters that contained areas with high ridge values (shown as white in Figs. 2d, 2e, and 2f). Morphological opening (erosion and dilation) was applied using a disk structuring element to remove noise from these clusters. This process shrunk and then grew the clusters, removing any spurious connections between neighboring clusters. The cluster closest to a predicted value of the fingertip was selected, and its mean was taken as the estimate of fingertip location.

Prediction played an important role in our algorithm, by selecting values for the algorithm (the extent of thresholding and the radius of the structuring element in morphological opening) in a principled way. Instead of fine-tuning these values to our specific video parameters, participant hand size/color, lighting situation, etc., the algorithm applied multiple values of thresholding and opening, and those that produced a

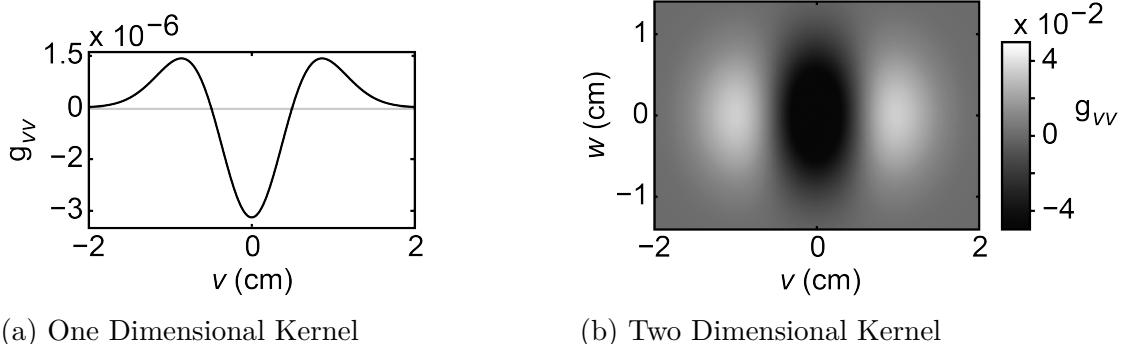


Figure 3. Second-order Gaussian kernel, g_{vv} ($\mu = 0$, $\sigma = 0.5$ cm), shown in one and two dimensions.

fingertip location estimate closest to the prediction were accepted. Predicted values were based on the linear combination (extrapolation) of estimates from the two previous video frames (mathematically defined in Figure 4). In addition to being used to select algorithm values (for thresholding and opening), the predicted location was used to select the cluster to associate with the index fingertip, and ignore spurious clusters or those belonging to the other fingertips.

Sometimes morphological opening was not enough to disconnect the fingertip cluster from clusters running along the length of the finger. Alternatively, sometimes in five-finger trials the clusters from the index finger and middle finger merged. These situations were easily detected based on the width/length ratio and the absolute width of the cluster. When the cluster was much longer than wide (5 times), it indicated that the cluster ran the length of the finger, and should be restricted to its top section (25% worked well). If the cluster was wider than a finger (2 cm) it indicated that the cluster belonged to adjacent index and middle fingers, and should be restricted to its left section (again, 25%). These heuristics could be changed in future implementations, depending on the video parameters, e.g., resolution. These values could also be chosen in the same manner that the algorithm chose values of thresholding and opening. The algorithm could be relatively agnostic, trying multiple values, ultimately using whichever values produced an estimated

fingertip location closest to the predicted fingertip location.

The complete algorithm using ridge detection for fingertip tracking is outlined in pseudocode in Figure 4. We implemented this algorithm in Matlab R2013b (The MathWorks, Inc., Natick, MA). Code is provided in the supplemental materials.

```

//Predict fingertip location,  $(\tilde{x}_t, \tilde{y}_t)$ , using
//previous estimates  $(\hat{x}_{t-1}, \hat{y}_{t-1})$  and  $(\hat{x}_{t-2}, \hat{y}_{t-2})$ 
if  $t > 2$  then
     $(\tilde{x}_t, \tilde{y}_t) \leftarrow (2\hat{x}_{t-1} - \hat{x}_{t-2}, 2\hat{y}_{t-1} - \hat{y}_{t-2})$ 
else if  $t == 2$  then
     $(\tilde{x}_t, \tilde{y}_t) \leftarrow (\hat{x}_{t-1}, \hat{y}_{t-1})$ 
else
     $(\tilde{x}_t, \tilde{y}_t) \leftarrow$  middle  $(x, y)$  of stimulus
end if
//Estimate fingertip location,  $(\hat{x}_t, \hat{y}_t)$ 
for various threshold values do
    for various radius values in opening do
        im  $\leftarrow$  image frame luminance values - ...
            background image luminance values
        Compute ridge detector  $L_{vv}$  by Equation 2;
        Apply thresholding, erosion, and dilation to  $L_{vv}$ 
        resulting in "clusters";
        Choose cluster that is closest to prediction
         $(\tilde{x}_t, \tilde{y}_t)$ ;
        if cluster is [5 times] longer than wide then
            //Cluster runs the length of the finger
            restrict cluster to its top [25%];
        else if cluster is wider than finger [2 cm] then
            //Cluster is merged index and middle
            restrict cluster to its left [25%];
        end if
         $(x', y') \leftarrow$  mean of cluster  $(x, y)$  values;
        if no existing estimate  $(\hat{x}_t, \hat{y}_t)$  or
         $(x', y')$  closer than  $(\hat{x}_t, \hat{y}_t)$  to  $(\tilde{x}_t, \tilde{y}_t)$  then
            // Take  $(x', y')$  as new estimate
             $(\hat{x}_t, \hat{y}_t) \leftarrow (x', y')$ 
        end if
    end for
end for

```

Figure 4. Algorithm implementation in pseudocode.

Application and Assessment of Algorithm

The 16 videos collected from each of the four participants yielded 26,040 one-finger video frames and 17,406 five-finger video frames. There were fewer video frames from five-finger trials because searching with one finger is slower than searching with five (Morash et al., 2013). Each video frame was processed by our algorithm, described in Figure 4. The estimates of fingertip position were visually checked, and rejected if the estimate was not correctly located on the index fingertip. Therefore, the algorithm was assessed based on three metrics: rejection rates and, for non-rejected fingertip position estimates, bias and variance.

To assess the bias (i.e., validity) and variance (i.e., reliability) of the algorithm estimates, we manually estimated the fingertip location for each frame. This was done twice by the same coder, who first estimated index fingertip locations for all frames with no feedback on the algorithm estimates, and then later re-estimated locations for all frames with no feedback on the previous manual or algorithm estimates. All visual inspection and manual estimation were completed using custom-made graphical user interfaces in Matlab.

Each estimate of fingertip location, either from the algorithm or manual estimators, was represented as the true fingertip location plus systematic bias, specific to the estimator, plus random error (Equation 3). The bias, or systematic error, of an estimator was the mean difference between the true fingertip location and the estimated fingertip location. That is, $\text{bias} = \mathbb{E}(d_{\text{true}} - d_{\text{estimate}})$, where d_{true} is a true fingertip location and d_{estimate} is the estimated fingertip location. In principle, we had no means to measure true fingertip location. Some statisticians may even argue that true fingertip position is only conceptually defined, is ill-defined, or does not exist. Therefore, we defined the manual estimates of fingertip locations to have zero bias. In other words, validity (bias) was assessed through comparison to the human-coder estimates.

We represented each fingertip estimate as $d_t = (x_t, y_t)'$, where t indexed the estimates

of the true fingertip location μ_t in video frame t . Then,

$$\begin{aligned} d_{t,m} &= \mu_t + \epsilon_{t,m} \\ d_{t,a} &= \mu_t + b + \epsilon_{t,a} \end{aligned} \tag{3}$$

where $d_{t,m}$ and $d_{t,a}$ were the two-dimensional $(x_t, y_t)'$ estimates from the manual and algorithm procedures, respectively. The algorithm's two-dimensional bias was $b = (b_x, b_y)'$. Lastly, $\epsilon_{t,m}$ and $\epsilon_{t,a}$ were the independent identically-distributed two-dimensional errors, with zero mean and variance-covariance matrices Σ_m and Σ_a , respectively. For brevity we will write $\Sigma_{d_m} = \Sigma_m = \mathbb{E}[(\epsilon_{x_m}, \epsilon_{y_m})'(\epsilon_{x_m}, \epsilon_{y_m})]$, and similarly $\Sigma_{d_a} = \Sigma_a$, $\Sigma_{d_m, d_a} = \Sigma_{m,a}$.

We assumed that $\Sigma_m = \Sigma_{m1} = \Sigma_{m2}$ (for the first and second manual estimates), and that neither these covariance-variance matrices, nor bias, were dependent on (x, y) , i.e., the bias and variance of the estimates were not larger in some areas of the map stimulus or video image than others. We also assumed that $\Sigma_{m2,m1}$ and $\Sigma_{a,m1}$ were both zero, i.e., due to independent ϵ_t 's in Equation 3 across measurements, there were no correlations in errors. Systematic error across manual estimates, which would typically be attributed to bias, were instead taken to be accurate fingertip locations (because bias was assumed to be zero), which the algorithm's bias was measured against.

Note that the algorithm's bias was defined as b , and the variance-covariance matrix as Σ_a , and these were the quantities we were interested in estimating. The two manual estimates provided a means to estimate these parameters. The manual estimates were referred to as $d_{t,m1} = (x_{t,m1}, y_{t,m1})$ for the first manual estimates, and $d_{t,m2} = (x_{t,m2}, y_{t,m2})$ for the second manual estimates, and these were setup in the following expressions to solve the algorithm's parameters.

$$\begin{aligned} \mathbb{E}(d_a - d_{m1}) &= b \\ \Sigma_{d_{m2}-d_{m1}} &= \Sigma_{m2} + \Sigma_{m1} - 2 \times \Sigma_{m2,m1} = 2 \times \Sigma_m \\ \Sigma_{d_a-d_{m1}} &= \Sigma_a + \Sigma_{m1} - 2 \times \Sigma_{a,m1} = \Sigma_a + \Sigma_m \end{aligned} \tag{4}$$

The variance-covariance relationships in Equation 4 relied on our assumption that $\Sigma_{m2,m1}$ and $\Sigma_{a,m1}$ were both zero, based on independence of ϵ_t 's.

We examined both estimate difference distributions, manual-manual: $d_{m2} - d_{m1}$ and algorithm-manual: $d_a - d_{m1}$, for bivariate normality using the Shapiro-Wilk multivariate normality test (Malkovich & Afifi, 1973). We also computed bivariate skew and kurtosis, and compared these to those expected from a bivariate normal distribution (Mardia, 1970). Statistical tests were done using R 3.0.2 (The R Foundation for Statistical Computing, Vienna, Austria).

Results

Rejection of Algorithm Estimates

For one-finger frames, 0.22% of frames had estimates that were deemed unacceptable by a human coder, because the estimate was not correctly located on the index fingertip. For the five-finger frames, 2.45% of frames were deemed unacceptable. These estimates were excluded from further analyses. We were unable to detect any specific trend or artifact that rejected frames had in common. The following bias and variance results were calculated using pooled estimates across all one- and five-finger trials, yielding a sample of 43,446 frames. Prior to pooling, we did not find any differences between one- and five-finger estimates, nor differences between participants.

Estimate Distributions

The distributions of differences between two manual estimates and between algorithm and manual estimates are shown in Figure 5 (only one algorithm-manual distribution is shown, because the other is similar and was not used in computation). The manual estimates revealed no bias (between them), the manual-manual distribution was centered on zero (Wilcoxon rank-sum test, x values $p = 0.39$, y values $p = 0.57$). The algorithm estimates were biased relative to the manual estimates, $b = (0.08, 0.04)$ cm, which was

significantly different from zero (Wilcoxon rank-sum test, x values $p < 0.001$, y values $p < 0.001$).

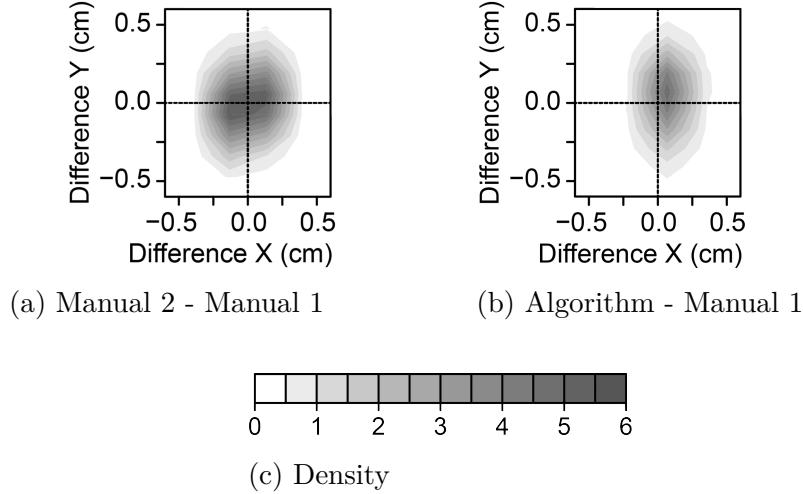


Figure 5. Distributions of estimate differences.

Both manual-manual and algorithm-manual distributions appeared roughly symmetric (Figure 5), however neither distribution was bivariate normal based on Shapiro-Wilk multivariate normality tests (manual-manual: $W = 0.98$, $p < 0.001$; algorithm-manual: $W = 0.99$, $p < 0.001$). Furthermore, a multivariate test of skew and kurtosis (Mardia, 1970) indicated a small skew in the manual-manual differences, $\beta_{1,2} = 0.02$, significantly different from zero ($p < 0.001$), and kurtosis $\beta_{2,2} = 24.47$, significantly different from normal $\beta_{2,2} = 8$, ($p < 0.001$). For the algorithm-manual differences, skew was $\beta_{1,2} = 0.26$ ($p < 0.001$) and kurtosis was $\beta_{2,2} = 14.62$ ($p < 0.001$). Excess kurtosis indicated that these distributions were more concentrated towards the mean, or more peaked, with fatter tails than normal. Marginal distributions are shown in Figure 6.

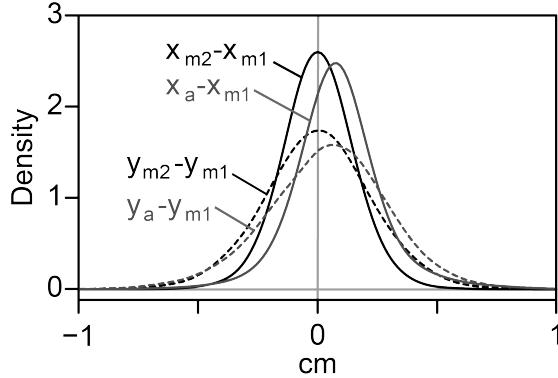


Figure 6. Marginal distributions of estimate differences.

The covariance-variance values for estimate differences are shown below, with standard errors (SEs) of the variance and covariance estimates shown in parentheses. SEs were estimated by nonparametric bootstrap (Monte Carlo case resampling), with 100,000 iterations and a sample size equal to the number of frames.

$$\Sigma_{d_{m2}-d_{m1}} = \begin{bmatrix} 24.06 (0.45) & 4.12 (0.21) \\ 4.12 (0.21) & 52.73 (0.43) \end{bmatrix} \times 10^{-3} \text{ cm}^2$$

$$\Sigma_{d_a-d_{m1}} = \begin{bmatrix} 37.24 (1.27) & 1.50 (0.98) \\ 1.50 (0.98) & 69.38 (1.71) \end{bmatrix} \times 10^{-3} \text{ cm}^2$$

Then, the following were computed using the above expressions and Equation 4, SEs shown in parentheses.

$$\Sigma_m = \begin{bmatrix} \sigma_{x_m}^2 & \gamma_{x_m, y_m} \\ \gamma_{y_m, x_m} & \sigma_{y_m}^2 \end{bmatrix} = \begin{bmatrix} 12.03 (0.32) & 2.06 (0.15) \\ 2.06 (0.15) & 26.37 (0.30) \end{bmatrix} \times 10^{-3} \text{ cm}^2$$

$$\Sigma_a = \begin{bmatrix} \sigma_{x_a}^2 & \gamma_{x_a, y_a} \\ \gamma_{y_a, x_a} & \sigma_{y_a}^2 \end{bmatrix} = \begin{bmatrix} 25.21 (0.29) & -0.56 (0.30) \\ -0.56 (0.30) & 43.02 (0.50) \end{bmatrix} \times 10^{-3} \text{ cm}^2$$

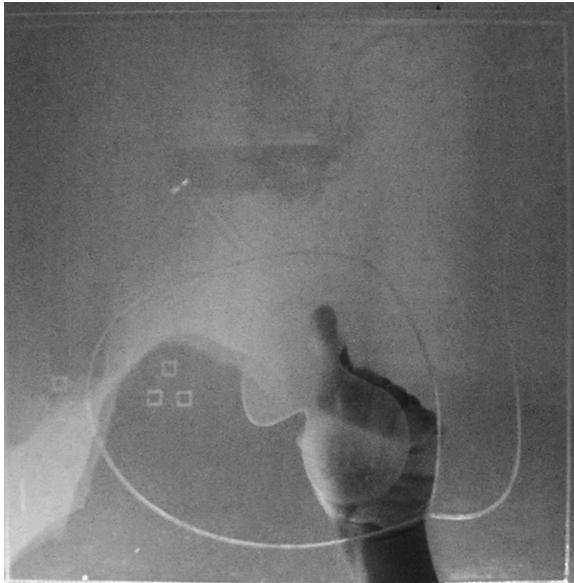
The covariance $\gamma_{x_a, y_a} = -0.56 \times 10^{-3} \text{ cm}^2$ was not significantly different from zero

($p = 0.081$), but the covariance $\gamma_{x_m,y_m} = 2.06 \times 10^{-3} \text{ cm}^2$ was ($p < 0.001$). The variance/covariance values can be rewritten as standard deviations and correlations ($\rho_{x,y} = \gamma_{x,y}/\sigma_x\sigma_y$) with more interpretable units of cm,

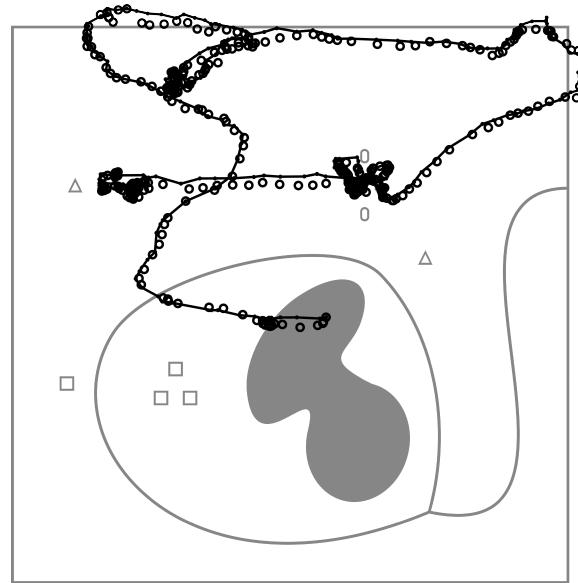
$$\sigma_{x_m} = 0.11 \text{ cm} \quad \sigma_{y_m} = 0.16 \text{ cm} \quad \rho_{x_m,y_m} = 0.12$$

$$\sigma_{x_a} = 0.16 \text{ cm} \quad \sigma_{y_a} = 0.21 \text{ cm} \quad \rho_{x_a,y_a} = -0.02$$

An example of the outcome of finger tracking, using both manual and algorithm procedures, is shown in Figure 7. The manual and algorithm estimates appear similar and mostly overlap.



(a) First Video Frame



(b) Manual (Line) and Algorithm (Circle) Tracking

Figure 7. Example of finger tracking on 421 video frames.

Example Applications of Finger Tracking for Haptics Research

This section provides some examples of finger tracking applications in perceptual psychology, which make explicit use of the finger-tracking variance and bias calculated above.

Positional Variation Explained by Index Finger. During five-finger haptic search, the hand may be held in a static pose, with the fingers kept a fixed distance from one another. Alternatively, the fingers may move more independently, causing whole-hand or local contractions and expansions (Morash et al., 2013). If the fingers move together, the position of one finger could be perfectly predicted by the position of another. This implies the regression equation, $f_1 = \alpha + \beta f_2 + \epsilon$, where f_1 and f_2 are the positions of the two fingers, α and β are constants, and ϵ is the residual error.

We regressed the positions of thumb, middle, ring, and little fingers onto the position of the index finger (using the algorithm to extract each finger's position). The explained variance, $1 - \text{Var}(\text{residuals})/\text{Var}(\text{total})$, could vary from 0, indicating no linear relationship, to 1, indicating a perfect linear relationship. However, due to the algorithm's variance, a value of 1 is impossible, and it may therefore be desirable to apply a correction for the algorithm's known variance.

Proportion variance explained was calculated for each trial, then averaged across trials within participant to form participants' means, which were used to create overall averages with SEs, shown in Table 1. No finger's position was completely explained by the index finger's position, even after correcting for the algorithm's variance. This suggests that the fingers' movements are at least partially independent during five-finger search, and the hand is not held in a static pose that is moved around the stimulus.

	Thumb	Middle	Ring	Little
<i>Uncorrected for Algorithm Variance</i>				
<i>x</i>	0.83 (0.05)	0.90 (0.04)	0.85 (0.06)	0.76 (0.08)
<i>y</i>	0.76 (0.07)	0.94 (0.01)	0.91 (0.01)	0.88 (0.01)
<i>Corrected for Algorithm Variance</i>				
<i>x</i>	0.84 (0.04)	0.92 (0.04)	0.86 (0.06)	0.77 (0.08)
<i>y</i>	0.77 (0.07)	0.95 (0.01)	0.92 (0.01)	0.89 (0.01)

Table 1

Average proportion of variance explained (with SEs) by the position of the index finger for other finger positions in five-finger search trials.

Index Finger Position on Stimulus Features. Visual search includes primarily two states of eye position: fixations on image features, associated with small fixational eye movements (microsaccades, etc.), and saccades, quick jumps between fixations (Bogartz & Staub, 2012). Therefore, the vast majority of gaze positions are located on task-relevant image features, and not in the space between image features, which are quickly passed over during saccades (Yarbus, 1967). To investigate whether finger locations are similarly located mostly on symbols (our stimuli's main features), we computed the percentage of index-finger locations that were within 0.75 cm (about half a fingertip width) of map symbols, shown in Table 2. Prior to this computation, we corrected for the fingertip-location bias by subtracting (0.08, 0.04) cm from each estimated location.

Participant	% Locations on Symbols	
	One-Finger	Five-Finger
1	12.44 (10.22)	1.01 (0.23)
2	6.17 (38.77)	0.57 (0.10)
3	5.56 (1.89)	0.93 (0.25)
4	5.10 (2.52)	0.65 (0.08)
Ave.	7.32 (1.72)	0.79 (0.11)

Table 2
Average (with SEs) finger positions within 0.75 cm of a map symbol.

A relatively small percentage of index finger positions were located within 0.75 cm of a map symbol, on average 7.32% for one-finger trials and 0.79% for five finger trials. Therefore, the majority of index-finger locations were located in “open space,” away from the map symbols. This was even more true in five-finger than one-finger searches, perhaps due to the exploration of symbols by non-index fingers.

Discussion

This report presents a method to assess the bias and variance of a deterministic finger-tracking algorithm. Through comparison of our algorithm's estimates to those from a variable method (human coding), we were able to determine our algorithm's bias and variance. We consider our approach of estimating bias and variance the greatest contribution of this work. Specifically, we have demonstrated how these statistics can be measured despite the fact that the finger-tracking method was deterministic and did not produce different estimates from the same input image/data. Although we accomplished this through comparison to a human coder, comparison to any method with known or measurable bias and variance would suffice. Therefore, future work may measure the bias and variance of a future finger-tracking method by comparing its estimates to those from the current report's algorithm.

Measurements of bias and variance can be used to compare different finger-tracking methods, such as our algorithm and human coding. The algorithm's estimates were biased to be slightly larger in both x and y directions relative to the manual coding estimates. The displacements towards the right and further away (towards the finger tips) from the participant were extremely small (0.08 cm and 0.04 cm, respectively) compared to the width of the index finger (1.5 cm - 2.0 cm). This bias was evident in the marginal distributions, Figure 6, where it can be seen that manual-manual differences were centered on zero, but algorithm-manual differences were centered on values above zero in both x and y . However, it cannot be determined whether this relative bias was ultimately due to bias in the algorithm's estimates or bias in the manual estimates relative to "true" fingertip position. It's possible that manual estimates were biased to be low and towards the left. However, we believe that human coders are a reasonable judge of valid fingertip position.

Differences in estimates, manual-manual and algorithm-manual, were contained within 1.0 cm in both x and y directions. This is well within the size of a fingertip. These distributions both had significant excess skew compared to a bivariate normal distribution.

The skew was small in the manual-manual distributions ($\beta_{1,2} = 0.02$), and not visually apparent in the marginal distributions (Figure 6). In contrast, the skew in the algorithm-manual differences was larger ($\beta_{1,2} = 0.26$), and evident in the marginal distributions (Figure 6). In particular, the y algorithm-manual values had a negative skew. This is reasonable, considering that low y estimates would occur along the length of the finger, but high y estimates would be above, and not on the finger at all. Therefore, relative to low y estimates, high y estimates may have been less likely and/or excluded because they were not on the finger.

Both manual-manual and algorithm-manual differences had kurtosis that indicated greater concentration (a taller peak) around zero than a normal distribution. This is not necessarily a concern, because greater concentration reflects greater agreement between the estimates. Lastly, the variance of the manual-manual marginal distributions was smaller than the algorithm-manual distributions (Figure 6), reflecting that algorithm estimates had larger variance (less reliability) than manual estimates.

As previously mentioned, the standard deviation of the manual estimates was smaller than the algorithm estimates. However, both were much smaller than the width of an index fingertip (1.5 cm - 2.0 cm). There was a significant correlation in manual estimate error, $\rho_{x_m,y_m} = 0.12$. This indicated that errors tended to be either too high and towards the right, or too low and towards the left, when done manually. This may have occurred due to biases in hand/arm movements by the human coder using a computer mouse to estimate fingertip positions. There was no significant covariance in the algorithm estimate error. This is potentially more desirable, and one reason why algorithm estimates of fingertip location may be favored over manual estimates.

A secondary contribution of this report is our development of a finger-tracking algorithm using ridge detection. In general, an algorithm for estimating fingertip locations is more tractable than manually estimating fingertip locations in thousands of video-frame images. Therefore, creating a finger-tracking algorithm increases the accessibility of finger

tracking for psychological study. Marker-based approaches may be undesirable in a psychological research application, because these approaches assume line-of-sight between camera and marker(s), and ask the participant to wear a marker or glove that may impede finger movements and perception. Our algorithm offers a less-invasive alternative for tracking fingertips, which requires no specialized hardware to implement.

Overall, our finger-tracking algorithm incorrectly located the index fingertip in only 0.22% of video frames for one-finger videos (where all fingers except the index finger were taped to the palm) and in 2.45% of five-finger videos. A different finger-tracking study, which used infrared markers on the index finger, lost roughly 2% of samples due line-of-sight disruption between the video camera and infrared marker (van Polanen et al., 2011). Our one-finger results were much better than this, and our five-finger results were comparable. This demonstrates the viability of our algorithm for psychological research, especially when the researcher wants to use a video-based method that does not require markers to be attached to the participant.

Our algorithm was relatively simple, and could be improved in multiple ways, including greater removal of irrelevant features from video-frame images and better prediction. Each of these avenues of improvement could be specific to the context of application. For example, the stimuli and experimental setting will affect the information contained in video-frame images that is disruptive to ridge detection. Furthermore, the task the participant is performing will affect the hand movements, favoring some forms of prediction over others (e.g., autoregression or movement towards attractors). Because our algorithm relied on predicted fingertip locations to both select algorithm parameters (amount of thresholding and opening) and to choose an estimate from candidate estimates, better prediction has the potential to improve the algorithm's performance. Tracking could be improved using Kalman or particle filtering, which are popular in gesture recognition and object tracking applications (e.g., Bretzner, Laptev, & Lindeberg, 2002; Gustafsson, 2010). These approaches may be informed by the bias and variance estimates presented

above. However, we consider the current simplicity of our algorithm a positive attribute, as it will be straightforward for psychologists to implement.

A limitation of our algorithm is that its parameters were selected to provide good performance with our specific stimuli and setup. This reduces the generalizability of the algorithm. However, the algorithm's parameters, such as the width (2 cm) to determine that a cluster represents two fingertips instead of one, can be easily changed. Gesture recognition research often focuses on performance in non-optimal situations, where the hand(s) may be in front of a visually cluttered background, with unfavorable lighting, and/or captured using a noisy camera (Oikonomidis, Kyriazis, Tzivanidis, & Argyros, 2013). In such conditions, algorithms need to have parameters that automatically adjust to preserve performance. However, this may be of less concern for finger-tracking applications, where the algorithm can be fine-tuned to perform in a highly controlled laboratory setting.

Finally, this report contains two example applications of finger-tracking results in perceptual psychology. In the first, we determined the proportion variance explained by the position of the index finger for the other fingers in five-finger search. This proportion ranged from about 0.75 - 0.95, in x and y directions. Although these values were relatively close to 1.0, which would indicate perfect linear prediction, there was still as much as 25% variance unexplained in some instances. To examine if this unexplained variance could be attributed to the finger-tracking variance, we corrected for this in x and y directions using the variances we estimated. Only 1% - 2% of the previously unexplained variance was absorbed by the finger-tracking variance, resulting in corrected proportion of variance explained values that varied between 0.77 - 0.95, in x and y directions. The smaller values tended to be those belonging to the thumb and little finger, while the positions of the middle finger were most highly explained by those of the index finger. This suggests that the thumb and little fingers may move in opposition to the index finger more frequently than the middle finger, perhaps due to expansions and contractions of the hand.

We also examined the amount of time spent in contact with important stimulus

features (map symbols) by the index finger in one-finger and five-finger search trials. In this analysis, we corrected for our algorithm's bias relative to human judgments. Although the bias of the finger-tracking algorithm was small, adjusting for this bias took the human estimates of finger position to be more valid (correct) than the algorithm estimates, as we believe that humans are good estimators of fingertip position. We found that participants spent relatively little time touching important stimulus features with their index fingers, on average about 7.3% and 0.8% of frames in one- and five-finger trials, respectively. More time was spent touching important features with the index finger in one-finger trials, possibly because the non-index fingers were also used to examine important features in five-finger trials.

Our initial results on finger movements during haptic search may, along with future results revealed through finger-tracking studies, be informative to both basic perceptual and translational research. Finger movements, like eye movements, are a fundamental component of perception and action. Understanding the function and characteristics of finger movements during haptic perception may also be critical for translational research, such as the creation of haptic feedback in robot-assisted surgery, for which hand movement patterns of novice and expert surgeons may inform system design and training (Nisky, Okamura, & Hsieh, 2014).

In conclusion, we have shown how the bias and variance of a deterministic finger-tracking method can be measured through comparison to a method with known or measurable bias and variance. We demonstrate this approach for a deterministic finger-tracking method using ridge detection. This finger-tracking algorithm was designed specifically for psychological application, and we have detailed the method sufficiently for it to be reproduced by other perceptual psychologists. We use the measured bias and variance of our algorithm to compare the approach to human coding, whereby its bias ($x = 0.08$ cm, $y = 0.04$ cm) and standard deviation ($\sigma_x = 0.16$ cm, $\sigma_y = 0.21$ cm) were demonstrably small compared to the size of a human fingertip. Finally, we provide results on the

proportion of variance explained by the index finger for the other fingers in five-finger search, and the amount of time spent touching important stimulus fingers with the index finger, correcting for finger-tracking variance and bias, respectively. These contributions to the science and methodology of finger tracking may ultimately lead to greater adoption of finger tracking in psychology, and better understanding of haptic perception and action.

Acknowledgments

This research was supported by the Institute of Education Sciences (IES) pre-doctoral training grant R305B090026, the National Science Foundation (NSF) graduate research fellowship program DGE1106400, the Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship program, and Sigma Xi. The opinions expressed belong to the authors and do not represent the views of the IES, NSF, DoD, nor Sigma Xi.

References

- Bogartz, R. S., & Staub, A. (2012). Gaze step distributions reflect fixations and saccades: A comment on Stephen and Mirman (2010). *Cognition*, 123(2), 325–334.
- Breidegard, B. (2007). Computer-based automatic finger-and speech-tracking system. *Behavior Research Methods*, 39(4), 824–834.
- Bretzner, L., Laptev, I., & Lindeberg, T. (2002). Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering. In *Proceedings from the IEEE International Conference on Automatic Face and Gesture Recognition* (pp. 423–428).
- Farinella, G. M., & Rustico, E. (2008). Low cost finger tracking on flat surfaces. In *Proceedings from the Eurographics Italian Chapter Conference* (pp. 43–48).
- Green, C., Mihic, A., Brien, D., Armstrong, I., Nikkel, S., Stade, B., ... Reynolds, J. (2009). Oculomotor control in children with fetal alcohol spectrum disorders assessed using a mobile eye-tracking laboratory. *European Journal of Neuroscience*, 29(6), 1302–1309.
- Gustafsson, F. (2010). Particle filter theory and practice with positioning applications. *IEEE Aerospace and Electronic Systems Magazine*, 25(7), 53–82.
- Hayhoe, M., & Ballard, D. (2005). Eye movements in natural behavior. *Trends in Cognitive Sciences*, 9(4), 188–194.
- Huey, E. B. (1908). *The psychology and pedagogy of reading*. New York, NY: Macmillan.
- Jacob, R. J., & Karn, K. S. (2003). Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. *Mind*, 2(3), 4.
- Kim, H., & Fellner, D. W. (2004). Interaction with hand gesture for a back-projection wall. In *Proceedings from the IEEE Computer Graphics International* (pp. 395–402).
- Kimmel, D. L., Mammo, D., & Newsome, W. T. (2012). Tracking the eye non-invasively: simultaneous comparison of the scleral search coil and optical tracking techniques in the macaque monkey. *Frontiers in Behavioral Neuroscience*, 6.

- Klatzky, R. L., Loomis, J. M., Lederman, S. J., Wake, H., & Fujita, N. (1993). Haptic identification of objects and their depictions. *Perception & Psychophysics, 54*(2), 170–178.
- Lindeberg, T. (1998). Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision, 30*(2), 117–156.
- Maggioni, C. (1995). Gesturecomputer - New ways of operating a computer. In *Proceedings from the International Workshop on Automatic Face and Gesture Recognition* (pp. 166–171).
- Malkovich, J. F., & Afifi, A. (1973). On tests for multivariate normality. *Journal of the American Statistical Association, 68*(341), 176–179.
- Mardia, K. V. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika, 57*(3), 519–530.
- Millar, S. (1988). An apparatus for recording hand movements. *British Journal of Visual Impairment, 6*(3), 87–90.
- Morash, V., Connell Pensky, A. E., & Miele, J. A. (2012a). *Tactile map open stimulus set*. Retrieved from <http://www.valeriemorash.com/tactilemaps/>
- Morash, V., Connell Pensky, A. E., & Miele, J. A. (2012b). The tactile map open stimulus set for tactile and haptic research. *Journal of Visual Impairment & Blindness, 106*(8), 501.
- Morash, V., Connell Pensky, A. E., & Miele, J. A. (2013). Effects of using multiple hands and fingers on haptic performance. *Perception, 42*(7), 759–777.
- Nisky, I., Okamura, A. M., & Hsieh, M. H. (2014). Effects of robotic manipulators on movements of novices and surgeons. *Surgical Endoscopy, 28*(7), 2145–2158.
- Oikonomidis, I., Kyriazis, N., Tzевanidis, K., & Argyros, A. A. (2013). Tracking hand articulations: Relying on 3D visual hulls versus relying on multiple 2D cues. In *Proceedings from ISUVR: IEEE International Symposium on Ubiquitous Virtual Reality* (pp. 7–10).

- Pavlovic, V. I., Sharma, R., & Huang, T. S. (1997). Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7), 677–695.
- Symmons, M., & Richardson, B. (2000). Raised line drawings are spontaneously explored with a single finger. *Perception*, 29(5), 621–626.
- ter Haar Romeny, B. M. (2003). *Differential structure of images*. Berlin: Springer.
- ter Haar Romeny, B. M. (2013). The differential structure of images. In V. Lakshminarayanan, M. Calvo, & T. Alieva (Eds.), *Mathematical optics: Classical, quantum, and computational methods* (pp. 581–597). Oxford: CRC Press.
- van Polanen, V., Tiest, W. M. B., & Kappers, A. M. (2011). Movement strategies in a haptic search task. In *Proceedings from the IEEE World Haptics Conference* (pp. 275–280).
- Wang, R. Y., & Popović, J. (2009). Real-time hand-tracking with a color glove. In *Proceedings from TOG: ACM Transactions on Graphics* (Vol. 28, p. 63).
- Yarbus, A. L. (1967). *Eye movements and vision*. New York, NY: Plenum Press.
- Zabulis, X., Baltzakis, H., & Argyros, A. (2009). Vision-based hand gesture recognition for human-computer interaction. In C. Stephanidis (Ed.), *The universal access handbook, human factors and ergonomics* (chap. 34). Boca Raton, FL: CRC Press.