Human Hand Modeling from Surface Anatomy

Taehyun Rhee*
University of Southern California

Ulrich Neumann†
University of Southern California

J.P. Lewis‡
Stanford University

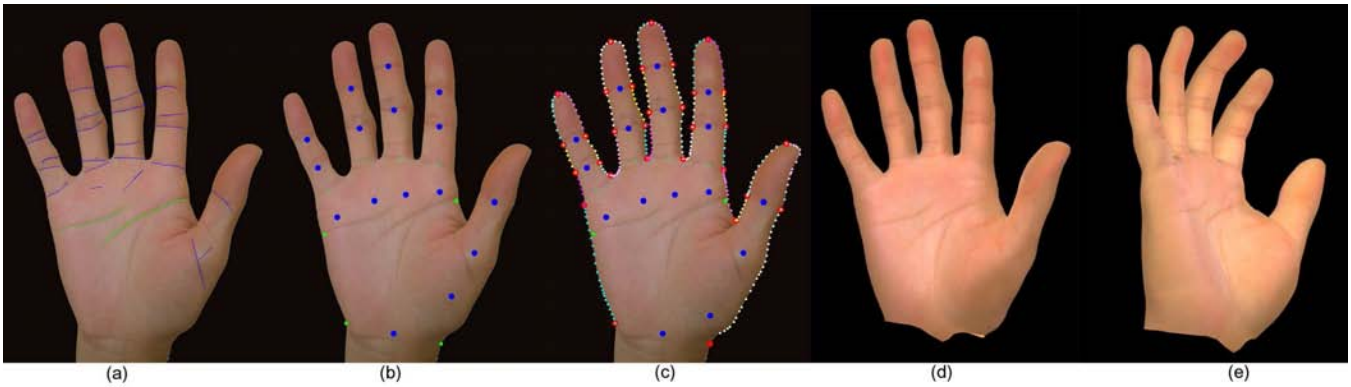

Figure 1: Human hand cloning from surface anatomy: (a) palm creases extracted using tensor voting and surface anatomy, (b) estimated hand joints, (c) feature correspondences; red dots represent curve segments, (d) cloned 3D model of female hand, (e) cloned 3D model in a different view and position with inherited skinning properties.

Abstract

The human hand is an important interface with complex shape and movement. In virtual reality and gaming applications the use of an individualized rather than generic hand representation can increase the sense of immersion and in some cases may lead to more effortless and accurate interaction with the virtual world. We present a method for constructing a person-specific model from a single canonically posed palm image of the hand without human guidance. Tensor voting is employed to extract the principal creases on the palmar surface. Joint locations are estimated using extracted features and analysis of surface anatomy. The skin geometry of a generic 3D hand model is deformed using radial basis functions guided by correspondences to the extracted surface anatomy and hand contours. The result is a 3D model of an individual's hand, with similar joint locations, contours, and skin texture.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations; I.4.6 [Image Processing and Computer Vision]: Segmentation—Edge and feature detection

Keywords: Graphics, Modeling, Vision, Human hand, Anatomy, Curve detection, Tensor voting

1 Introduction

The hand is a very important interface for humans. Many interactions are performed by hand including object handling, communicating, and numerous other tasks in everyday life. The human hand needs to play a similar role in the virtual world in order to allow a more immersive and accurate interaction within a virtual environment. A realistic and accurate virtual hand model may be required for applications in virtual reality, virtual prototyping, animation, special-effects, games, ergonomics, and medical simulation. However, modeling an accurate and realistic virtual human hand is difficult and requires great skill since the human hand has a complex shape with many degrees of freedom (DOF); an elaborate system of joints and muscles for control, and subtle skin deformation arising from the motion of muscles and tendons.

1.1 Related work

The subject has been considered in human computer interaction (HCI), gesture recognition, medical simulation, as well as in several important graphics papers. Simple hand models have been developed for gesture interface research [Wu and Huang 1999] and research related to motion analysis, hand tracking, and gesture recognition. In vision-based tracking and analysis, accurate kinematical models and joint constraints are required based on biomechanical and anatomical hand motion analysis, but a hand model with simple geometry and without skin deformation is used in general [Lin et al. 2000].

Thompson et al. described an interactive 3D graphics workstation system for hand surgery practice that encompasses the simulation of both the kinematics and dynamics of the human hand [Thompson et al. 1988]. A cadaver hand specimen was used to obtain CT images, from which 3D structures were modeled. The segmentation of each bone and joint axis was accomplished by manual input in an interactive 3D display workstation.

In addition to an accurate kinematics model, high quality hand geometry and skin deformation models are desirable in computer graphics applications. Moccozet et al. pursue hand modeling and animation based on Dirichlet free-form deformations (DFFDs) [Moccozet and Magnenat-Thalmann 1997]. A multi-layer deformation model simulates the intermediate layer between the skeleton and the skin. Skin deformations use the palm and finger creases as constraints, based on the observation that each joint of the skeleton

*e-mail: trhee@usc.edu
†e-mail: uneumann@graphics.usc.edu
‡e-mail: zilla@computer.org

I3D 2006, Redwood City, California, 14–17 March 2006.
© 2006 ACM 1-59593-295-X/06/0003 $5.00

is associated with a hand crease on the surface. However, in their paper, the creases were determined and designed manually.

Kry et al. use a finite element model (FEM) of the human hand to obtain models in an example-based approach. The models are then compressed using principal component analysis, and realtime deformation is demonstrated using GPU acceleration. [Kry et al. 2002].

Albrecht et al. developed a human hand model with its underlying anatomical structure: skin, muscles, and bones [Albrecht et al. 2003]. Hand animation employs a hybrid muscle model. Pseudo muscles control the rotation of bones based on anatomical analysis and mechanics. Geometric muscles control the deformation of the skin using a mass-spring system. The hand skin geometry is based on a 3D scanner and the bone-joint structure is obtained from a public 3D skeleton model. Anatomical analysis and physical models achieve accurate finger movements and skin deformations.

In [Kurihara and Miyata 2004] an example-based deformable human hand model is derived from medical images. Multiple CT scans of the hand are taken in several poses. A comparison of bone shapes and different poses allows the estimation of joint locations. Hand skin deformation is achieved using the weighted pose space deformation (PSD) method.

Tsang et al. describe an anatomically accurate skeletal musculotendon model of human hand. The hand skeleton can be moved using muscle activations which derive forward and inverse dynamics simulation [Tsang et al. 2005].

Determination of realistic hand poses for interacting with other objects is also a challenging research topic in computer graphics and robotics. Several papers suggest interesting solutions for this issue [Huang et al. 1995; Kim et al. 2000; ElKoura and Singh 2003; Pollard and Zordan 2005].

It may be noted that the general approach of deforming a generic prior model to match individual shape data has been successfully applied to heads [Kähler et al. 2002] and the whole body [Allen et al. 2003; Seo et al. 2003], and that problem domain knowledge was usefully applied in each of these cases.

1.2 Motivations

Virtual reality, simulation, and entertainment applications often display a proxy of the operator's hand in the virtual world. The virtual hand may be driven by camera tracking, a haptic interface, or a sensor/actuator globe, and serves to provide visual feedback on object interaction as well as enhancing realism. The operator is intended to regard this virtual hand as *their* hand for the purpose of interacting in the virtual world. Differences in shape and mechanics between the operators' real and virtual hands may therefore cause inaccuracies in simulations and decrease the sense of immersiveness. Unfortunately, due to the difficulty, time, and cost of generating a person-specific hand model, immersive systems today generally use only a simple generic hand model.

Our goal is to make a realistic person-specific 3D virtual hand model, including skin geometry, texture, and the underlying skeleton. Our process requires minimal human guidance and depends only on capturing a skin texture image whose viewing direction is perpendicular to the palm plane. By analysis of the skin texture of the hand, we extract surface anatomy information and estimate osseous structure underneath the skin.

Our work is most similar to that in [Albrecht et al. 2003], which employs an image-based hand deformation method to generate individual hand models. Their method uses an image and scattered interpolation to deform a generic hand model, but it requires manually defined correspondences between 3D feature points and the hand image. Furthermore, the limited numbers of feature correspondences are not enough to generate a completely accurate shape match between the source picture and the final 3D hand model. We improve upon their efforts in several ways described next.

1.3 Overview and Contributions

We present an automated method to make a specific human hand model from an image of the surface anatomy of a human hand. The method has two main parts.

The first part is a surface anatomy feature-extraction method, based on tensor voting, to extract the main creases on the palmar skin and the hand geometry without human guidance. Joint structure is estimated from an anatomical analysis of the relationships between the surface anatomy and its osseous structure.

The second part deforms a predefined generic 3D hand model using scattered data interpolation based on a radial basis functions (RBFs). A curve segment matching method performs automatic feature correspondences between the 3D hand model and the contours detected in the hand image. These matches capture the contour of the hand image as well as the geometry of joints, finger tips and finger valleys.

As our contribution is the modeling of person-specific 3D hand geometry, issues such as animation of skin deformation, texture capture, and texture blending are beyond the scope of the paper, and we rely on existing techniques for these problems. Our generic model is manually generated using Maya, and the person-specific hand inherits the skin deformation system of the generic hand. Alternatively sophisticated automated techniques for animated skin deformation such as [Kurihara and Miyata 2004; Kry et al. 2002] could be adapted.

We texture mapped the hand using planar projection and manually blended the palm and dorsal textures. However improved texture blending techniques such as [Burt and Adelson 1983; Zhou et al. 2005] should be considered.

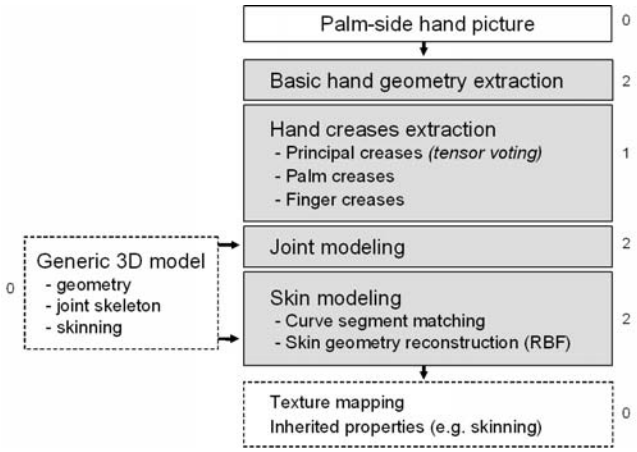

Figure 2: **Process overview:** Two main parts are grouped into four steps in the gray boxes. Supplementary parts which need additional tools or are beyond the scope of our work are indicated by the dotted boxes. Numbers beside each box indicate the required user interaction level; level 2 is fully automatic, level 1 requires no user interaction but needs some parameters (experimentally fixed in normal case; Canny edge thresholds, tensor voting σ, finger extraction thresholds), level 0 needs manual input.

As a result, the final hand model has the same joint structure, hand geometry, contour curves, and texture map as the original hand image. Our method requires only an image of a hand and requires no manual input. Figure 2 describes our methods and user interaction level in each step. The details are covered in the remainder of the paper.

2 Hand anatomy

The human hand has a complex anatomical structure consisting of bones, muscles, tendons, skin, and the complex relationships between them [Brand and Hollister 1999; Kry et al. 2002]. Analysis of human hand anatomical structures is important in various fields, including ergonomics, HCI, hand surgery, as well as computer animation.

The bones of the hand are grouped into three areas: digital, carpal, and wrist. The digital bones of four fingers consist of distal, middle, and proximal phalanges with distal interphalangeal (DIP), proximal interphalangeal (PIP), and metacarpal phalangeal (MCP) joints. The digital bones of the thumb consist of the distal and proximal phalange with the interphalangeal (IP) and carpometacarpal (CMC) joints. The carpal bones are a set of eight small bones, and the wrist bones are the radius and ulna [Figure 3(a)].

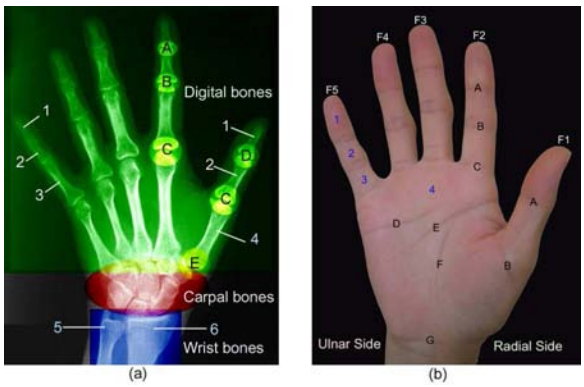

Figure 3: **Human hand anatomy: (a) Bone Anatomy** 1. Distal phalanx, 2. Middle phalanx, 3. Proximal phalanx, 4. Metacarpals, 5. Ulna, 6. Radius, A. DIP joint, B. PIP joint, C. MCP joint, D. IP joint, E. CMC joint; **(b) Surface Anatomy** 1. Distal phalanx, 2. Middle phalanx, 3. Proximal phalanx, 4. Distal palmar, A. DIP crease, B. PIP crease, C. Palmar digital crease, D. Distal palmar crease, E. Proximal palmar crease, F. Thenar crease, G. Distal wrist crease, F1-5. Thumb, Index, Middle, Ring, Little finger.

In biomechanics, not only the anatomical features but also the mechanical joints of the hand are important. Several methods are suggested to define the mechanical joints. In [Brand and Hollister 1999; Kurihara and Miyata 2004], the axis of rotation between two bones is defined as a line that does not move in relationship with either bone while the bones move around each other. On the other hand, the mechanical joint centers of the hand have been anatomically estimated as the center of curvature of the head of the bone proximal to the given joint [Buchholz et al. 1992; Youm et al. 1978].

Figure 3(b) shows the basic surface anatomy of the human hand [Yu et al. 2004]. The three palmar skin regions are the palm, fingers, and thumb. The palm region has three main creases. The proximal palmar crease starts from the silhouette edge of the hand near the head of the metacarpal bone of the index finger and runs through the hollow of the palm. The distal palmar crease starts from the head of the metacarpal bone in the little finger and runs through the hollow, passing the head of the metacarpal bone of the ring and middle finger. The thenar crease is located between the proximal palmar crease and distal wrist crease longitudinally.

The fingers have three transverse digital creases, the DIP, PIP, and palmar digital crease. A DIP crease and palmar digital crease are located on the thumb. The distal palm area is located between the line, which starts from the radial end of the proximal palmar crease to the ulnar end of the distal palmar crease and the palmar digital creases. The MCP joints of fingers are located in the distal palm area.

Since the crease on the palmar skin is produced by a skin flexion fold when the hand is closed, basic creases on the palmar skin have a strong relationship with the underlying bone structure, resulting in landmarks used in hand surgery [Brand and Hollister 1999; Yu et al. 2004]. Bugbee et al. demonstrated the relationship between creases of palmar skin and superimposed osseous anatomy using radiograph [Bugbee and Botte 1993]. Since most skin creases of the palmar skin are consistent with underlying fascia and located near the center of the curvature of the head of the bone proximal of the given joint, we observe that the creases can be used as an estimation of mechanical joints when folding the human hand.

The surface anatomy of the hand, as well as the bone structure, is unique for an individual human hand [Yu et al. 2004]. Due to their uniqueness, hand surface features have been proposed for biometric use. For example, a palm print has three principal lines (distal palmar, proximal palmar, and thenar creases) that are unique and unchanging biometrics suitable to identify a person [Shu and Zhang 1998; Jain et al. 1999]. Also, palmistry uses these lines to indicate individual characteristics.

3 Basic hand geometry extraction

In this paper, we define basic hand geometry as the contour of the hand, finger tips, and finger " valleys " shown in Figure 4(b). Since we use skin texture as the base hand image, it is generally taken within good lighting conditions, simple dark background, fingers-extended natural pose, and fixed camera location. The camera viewing direction is perpendicular to the palm plane and the image up vector is on the axis from wrist joint to middle finger tip. We assume an orthographic camera and explain our method based on the right hand due to the similarity and symmetry of the left and right hand.

Generally, contour extraction is a similar problem as background extraction in computer vision. In our case, since we used a specified background, a simple Canny edge algorithm [Forsyth and Ponce 2002] followed by a binary image thresholding is enough to extract a proper contour line [Figure 4(b)]. When we see the surface anatomy of each finger, their shapes are almost symmetrical and each finger tip is located on the intersection between the medial axis curve of each finger and the hand contour curve. Since our canonical photograph pose has the fingers oriented upward, we can easily locate the finger extremities by scanning the medial axis curves in the y-direction. When we sort the fingers with respect to x-axis, we can easily identify the finger tips for each finger. The finger valleys are located in between each finger. We trace a contour curve from each finger tip and define each finger valley as the inflection point of that curve.

4 Hand crease extraction

The palmar skin has complex geometry with lots of discontinuous wrinkles. Among the wrinkles, just a few salient creases are meaningful in terms of surface anatomy. Although humans can easily

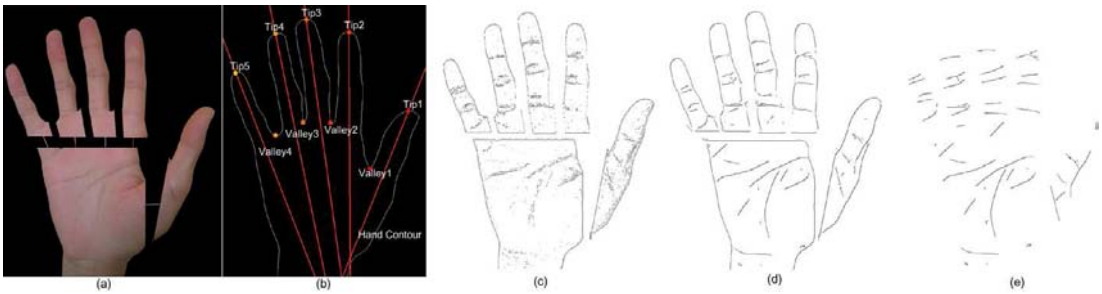

Figure 4: Palm creases extraction process: (a) Segmented hand image, (b) Basic hand geometry; contour of the hand, five finger tips, four finger valleys, and medial axis of each finger, (c) Sobel edge image, (d) First tensor voting result, (e) Second tensor voting result (final result)

detect these creases, computers can hardly detect them without human visual perception. Tensor voting [Guy and Medioni 1996] is an efficient perceptual grouping framework to detect dense salient features in noisy input data. We use the tensor voting framework [Medioni et al. 2000; IRIS] to detect principal creases and extract continuous curves for each crease in the palmar skin image.

4.1 Tensor voting overview

Extracting main creases from the palmar skin is a similar problem to inferring salient curves from sparse and noisy data. Tensor voting can extract salient geometric features (e.g. point, curve, surface) in a sparse and noisy data set without human guidance. The tensor voting framework proposed in [Medioni et al. 2000] needs only one free parameter σ to define the scale of the vote, and handles multi-dimensional data as well as 2D data for various applications. Tensor voting can be grouped into two major stages: data encoding using tensors and linear tensor voting for tensor communication. The input tokens, a set of points or oriented points in a noisy environment, can be encoded into a set of geometric features which are salient, smooth and considerably continuous.

In the case of 2D, each token can be encoded into a second order symmetric tensor which is equivalent to an ellipse whose major axis serves as the orientation and the length of that axis is the saliency. In Figure 5(a), an oriented point is represented by a thin oriented ellipse. In the voting stage, each token is accumulated into the second order symmetric tensor form by votes cast from its neighbors' predefined voting fields. The voting field of oriented tokens can be derived from the 2D stick tensor (degenerate ellipse with $\lambda_2 = 0$) called the fundamental 2D stick kernel, and it decays the saliency of the vote. In spherical coordinates, the decay of the 2D stick kernel is of the form:

$$DF(\gamma,\ \theta,\ \sigma) = e^{\frac{-(\ \gamma^2 + c\theta^2)}{\sigma^2}} \tag{1}$$

Where, γ is the arc length between voter and receiver, θ is the curvature between voter and receiver, σ is the scale of voting which defines the size of the neighborhood of each token, and c is a constant. In the Figure 5(b), the osculating circle connecting voter O and receiver P generates the smoothest circular path connecting O and P, and the length of normal vector P represents the saliency of the vote. After voting, the second order symmetric tensor T can be decomposed into its eigenvectors e_1, e_2 with related eigenvalue $\lambda_1 \geq \lambda_2$.

$$T = [e_1, e_2]\begin{bmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{bmatrix}\begin{bmatrix}{e_1}^T\\ {e_2}^T\end{bmatrix} = \lambda_1 e_1 e_1^T + \lambda_2 e_2 e_2^T \tag{2}$$

Equation 2 can be rearranged with T_c, which represents curve component, and T_p, which represents point component encoding junction information.

$$T = (\lambda_1 - \lambda_2)e_1 e_1^T + \lambda_2(e_1 e_1^T + e_2 e_2^T) = T_c + T_p \tag{3}$$

As a result of voting in 2D space, we have two dense vector maps which represent curve and junction features. The salient curves are located at the local extrema of map T_c, and they can be extracted using non-maxima suppression. Please refer to [Guy and Medioni 1996; Medioni et al. 2000] for details.

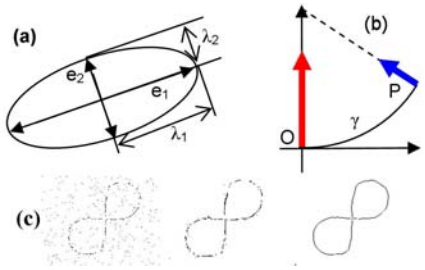

Figure 5: Tensor Voting: (a) tensor ellipse, (b) 2D stick kernel, (c) Tensor voting example: left image shows the sparse input tokens without orientation, middle image is the tokens oriented by tensor voting but still sparse, the right image is the resulting extracted dense curve.

4.2 Wrinkle extraction using tensor voting

In order to increase the performance of tensor voting, we segment the hand image. From the hand geometry extracted in section 3, we can roughly divide hand images into three regions; the palm, thumb, and four fingers. In the four fingers and thumb region, the medial axis for each fingers and thumb are calculated. For all pixels in the four fingers and thumb region, find the nearest medial axis line. The region which is related to the nearest line of the given pixel is assigned to the region for that pixel. Finally we achieve six image segments, like Figure 4(a). The image segments are converted into the set of edge points using the Sobel edge algorithm [Forsyth and Ponce 2002]. Although the Canny edge algorithm is generally used for edge detection, the Sobel edge algorithm shows better performance as an input of tensor voting in our experiment [Figure 4(c)].

Since the only free variable σ is not sensitive, we experimentally define the voting scale σ as large as possible to reduce the influence of local features which can produce noise. The result of first tensor voting is in Figure 4(d). In order to eliminate small noises and achieve smoother curves, we perform a second tensor voting using the input of the first tensor voting results. In the second voting stage, the outline of the fingers which are generated by the first tensor voting is eliminated before voting, since the contour line is salient and can cause errors in the voting process. The final result

of tensor voting is the set of salient wrinkle curves and each curve is smooth and mostly continuous [Figure 4(e)].

4.3 Palm crease extraction

Now we have images of continuous principal wrinkles of the palm and each of the fingers but they do not yet correspond to the important creases of surface anatomy. In the palm region, we have three anatomically meaningful creases, the distal palmar, proximal palmar, and thenar crease [Figure 3]. Since the wrinkle curves achieved from tensor voting are continuous and have longer length than other curves, we can extract these creases using a tree search algorithm that employs knowledge of hand geometry.

Variables: R=region, r=start point, W=wrinkle, P=path, L=path length

1 Define region R using basic hand geometry [Section 3];
2 Scan R finding start points r_i of wrinkle curve W_i;
/* Where $r_i = \{r_1, ..., r_n\}, W_i = \{W_1, ..., W_n\}, r_i \in R$*/;
3 **for** *all* r_i **do**
4 | **while** *Traverse curve from* r_i *to the end using DFS* **do**
5 | | **if** *found the largest path* P_i **then**
6 | | | $W_i \longleftarrow P_i$; $L_i \longleftarrow \text{Length}(P_i)$;
7 **if** *IsMaximum*(L_i) **then**
8 | Crease C $\longleftarrow W_i$;
9 Repeat step 1 to 8 for each anatomical region R;

Algorithm 1: Palm crease extraction algorithm

First, we assign anatomical region R to find the starting point of the crease. For example, the root of the distal palmar crease can be located in region $R_d (R_d \in R)$ in Figure 9(a) (Color Plate). The region R_d can be defined easily using hand geometry features such as finger valley 4 and the MCP joint of thumb. The MCP joint of thumb can be roughly estimated using the finger line of thumb and finger valley 1. The region for the root of the proximal palmar crease can be assigned by similar anatomical analysis. Since the start points of several wrinkle curves can be in the region R, we should find the main crease among these wrinkle curves. First, we extract every wrinkle curve and its arc length using depth first search (DFS), since our wrinkle curve has no loops. For every start point r_i ($r_i \in$ region R), traverse wrinkle curve using DFS and assign the largest path P_i as the wrinkle curve W_i having length L_i. The wrinkle curve W_i which has the largest length among L_i is assigned to the crease C within the area R. We summarize the method in Algorithm 1. Green curves in the Figure 9(a) (Color Plate) are the extracted proximal palmar crease and the distal palmar crease.

4.4 Finger crease extraction

We propose a method to extract the crease of each finger from the given wrinkle images. First, label in turn each segmented image in section 4.2 as finger region R_i, (i for each finger). Then, assign finger line F_i using the least square line fit of the medial axis curve of each finger region R_i (calculated in section 3). For each finger, perform the following operations in Algorithm 2 to extract representative lines of finger creases. The result is in Figure 9(b) (Color Plate).

Variables: R: finger region, F: finger line, p: current position, W: wrinkle curves, O: orthogonal line(F, p), O': Selected lines, C: Selected crease line, threshold1 and threshold2 are defined after experiment.

1 **for** *each finger region* R_i *and line* F_i **do**
2 | **while** *Trace* F_i *from finger tips to wrist using* p_i **do**
| | /* $p_i (1 \le i \le 5,\ p_i \in R_i)$ is a current tracing position */;
3 | | **if** p_i *is an intersection point between* F_i and $W_{ik} = \{W_{i1}, ..., W_{in}\}$ **then**
4 | | | Calculate a line $O_{ik} = \{O_{i1}, ..., O_{in}\}$, which is orthogonal to F_i and passes through the intersection point;
5 | | **for** *all pixels* $X_i \in$ *creases of* R_i **do**
6 | | | **if** $|X_i - O_{ik}| <$ threshold1 **then**
| | | | vote(O_{ik});
7 | | **if** vote(O_{ik}) > threshold2 **then**
| | | $O'_{ik} \longleftarrow O_{ik}$, where $O'_{ik} = \{O'_{i1}, ..., O'_{im}\}$;
8 | | **if** O'_{ik} *is verified by the hand geometry such as finger region, tips, and valleys* **then**
| | | $C_{ik} \longleftarrow O'_{ik}$, where $C_{ik} = \{C_{i1}, ..., C_{iN}\}$;

Algorithm 2: Finger crease extraction algorithm

5 Hand modeling

We made a generic 3D hand model [Figure 6(c),(e)] which has skin geometry, joint skeleton, and skin deformation using Maya [Alias] and Poser [CuriousLab]. The skin geometry is exported from Poser and bound to the joint skeleton using Maya for skin deformation and convenient joint control. The joint skeleton has accurate biomechanical structure using 21 joints (including finger tips and wrist) [Figure 6(e)].

Each joint has a different degree of freedom for better control. Although carpal and wrist bones consist of several bone segments, since the radio carpal joint plays the most important part of overall movement, we defined one three-DOF mechanical joint for the wrist joint. DIP and PIP joints of each finger and the IP and MCP joints of thumb have one DOF. MCP joints of each finger have two DOF for the simplicity of control and the CMC joint of the thumb has three DOF due to its complexity. For simple notation, we will call the generic 3D hand model a generic hand, the 2D hand features extracted from the picture a source hand, and the remodeled 3D hand model a clone hand. The generic 3D hand model is reshaped as a clone hand that has the same mechanical joints, contour curve, and texture map as the unique human hand.

5.1 Joint modeling

From the careful analysis of hand surface anatomy (consisting of the basic hand geometry and creases extracted in the previous section), we can estimate mechanical joint locations of the source hand. From the finger crease lines C_{in} and finger lines F_i obtained in section 4.4., we can find DIP and PIP joints of the four fingers and the IP and MCP joints of the thumb. Since the finger lines are the medial axis, the mechanical joints are located on the intersection points between the crease lines C_{in} and finger lines F_i. If two creases are extracted near a joint, we locate the joint at the lower crease with the analysis of surface anatomy.

When we consider smooth skin deformation with crease constraints, MCP joints of the the four fingers should be located in the distal palm area [Figure 9(c) (Color Plate)]. From each finger line F_i we can calculate line segment S_i, which is between the related digital crease and the line H, which is defined by start points

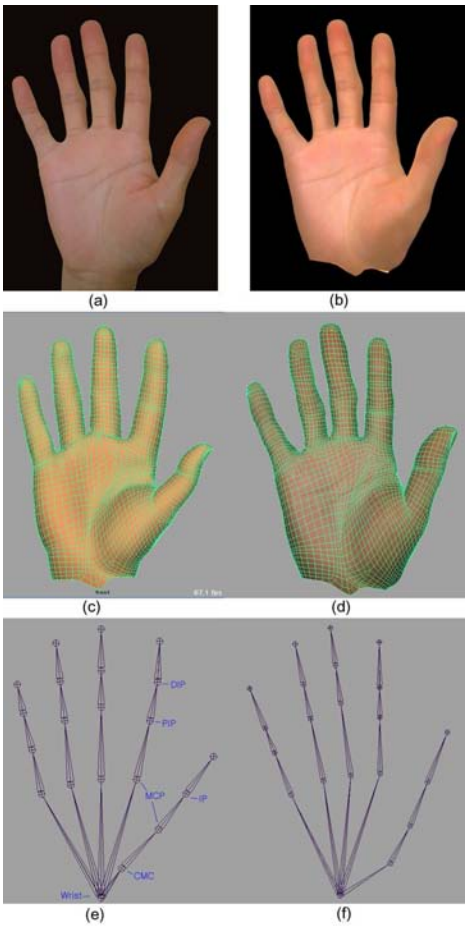

Figure 6: Hand clone results: (a) picture of source hand, (b) clone hand model with texture mapping, (c) skin geometry of generic hand model, (d) deformed skin geometry of clone hand, (e) joint skeleton of generic hand model (each circle is a joint location), (f) modified joint skeleton of clone hand

of distal palmar crease H_a and the start point of proximal palmar crease H_b [Figure 9(c) (Color Plate)]. Within each segment S_i, we can arrange MCP joints at approximately one-third of S_i from the line H with the analysis of surface anatomy in section 2.

The two highest curvature points, W_a and W_b of the ulnar and radial side of contour curve in wrist area, can represent the anatomical boundary between the wrist bones (ulna and radius) and carpal bones. These two points, W_a and W_b, are simply calculated by the highest curvature points of contour within the wrist region, which is below the MCP joint of the thumb. The wrist joint can be estimated by mid-position of W_a and W_b [Figure 9(c) (Color Plate)].

We have extracted every mechanical joint except the thumb CMC joint with analysis of surface anatomy. However, since the CMC joint of the thumb has relatively complex movements compared with other joints, and 2D surface anatomy cannot give enough information, estimating the CMC joint is very difficult. We position the thumb CMC joint based on the following anatomical analysis. First, the metacarpal bone of the thumb is parallel to line M and starts from the MCP joint of thumb [Figure 9(c) (Color Plate)]. Line M can be calculated by least square line fitting on the middle part of contour curve segment between the highest curvature point W_b and the MCP joint. The axis of the radius bone is parallel to the medial axis of wrist and passes point W_b. Since the anatomical CMC joint is located on the end of the metacarpal bone, the mechanical CMC joint can be on the intersection point between the axis of the radius bone and metacarpal bone.

From the extracted 2D joints, we can modify the joint skeleton of the 3D generic hand. Since the origin and the y-axis of the generic hand model is determined by the wrist joint and the vector from wrist joint to middle finger tip, respectively, a simple affine transformation matrix M_s can transfer the source hand coordinate to the generic hand coordinate. For accurate scaling, we measured hand length from the wrist to the middle finger tip when taking the hand picture. The generated joint skeleton model for clone hand is shown in Figure 6(f).

5.2 Skin modeling

In addition to the joints, the contour of the hand is a unique and important feature of a person's hand. Curve segment matching can allow automatic feature correspondence between the source hand and generic hand. From the corresponding features we can warp entire geometry using scattered data interpolation based on the radial basis functions (RBFs).

5.2.1 Curve segment matching

From the joint modeling process in section 5.1, we can obtain a joint skeleton model with 21 corresponding features. The skin vertices related to the joint skeleton can be used as the corresponding feature between the generic and source hand. In addition to these 21 features, contour of the hand provides good feature correspondences, since the source hand picture is also a skin texture image taken in the specific viewing direction and pose [Figure 6(a)].

When we deform the hand skin, the mesh between creases should be continuous. The contour of the hand can be divided into 30 curve segments $G_i = \{G_1, ..., G_{30}\}$ based on the hand creases and geometry from the analysis of hand surface anatomy [Figure 1(c)]; each G_i consists of ordered 3D vertices $V_{ij} = \{V_{i1}, ..., V_{in}\}$ along the curve segment G_i, where n is chosen to represent curve segment geometry sufficiently and G_i are carefully extracted when making a generic hand model using the related joint skeleton.

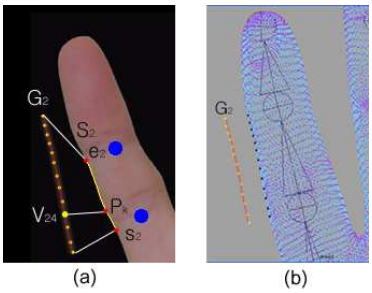

Figure 7: Curve segment matching example: (a) Curve matching example of curve segment $S_2(s_2, e_2)$: s_2 is on the intersection between contour and the palmar distal crease of little finger, and the e_2 is on the intersection between contour and the PIP crease of little finger, (b) Curve segment 2 in generic hand model

Every G_i is transformed to the source hand coordinate using the inverse matrix of M_s in section 5.1. The transformed curve segments are $G'_i = \{G'_1, ..., G'_{30}\}$ and each segment G'_i consists of transformed 2D points V'_{ij}. With curve segment matching based on the arc length, we can find corresponding feature of V'_{ij} in the source hand image. Our algorithm can accumulate small errors within its traversing step but it shows acceptable results in our experiment.

The method is described in Algorithm 3 and an example of curve segment 2 is in Figure 7.

Variables: G: curve segments of the generic hand, G': transformed curve segments of the generic hand, S: curve segments of the source hand, V': transformed vertices composing a curve segment G'

Define curve segment $S_i(s_i, e_i) = \{S_1, ..., S_{30}\}$ from hand anatomy [Figure 1(c)];
/* s_i: start point of segment, e_i: end point of segment */;
/* Each G'_i correspond with segment S_i using its anatomical location */;
Calculate arc length of G'_i and S_i;
Calculate scaling γ = length(S_i)/ length(G'_i);
for *all segment i* **do**
    for *all vertices j in segment i,* $V'_{i,j} = \{V'_{i,1}, ..., V'_{i,n}\}$ **do**
        Calculate $D_j = |V'_{i,j+1} - V'_{i,j}|$ /* n: number of vertices on the G'_i */;
    Initialize $s'_j = p_k = s_i$; /* s'_j : current start point, p_k : pixels in the S_i */;
    while *traverse* p_k *from* s_i *to* e_i **do**
        Calculate $D'_j = |p_k - s'_j|$;
        if $D'_j \geq \gamma D_j$ **then**
            Assign p_k as the corresponding feature of $V_{i,j+1}$;
            $s'_j \longleftarrow p_k$;

Algorithm 3: Curve segment matching algorithm

5.2.2 Skin deformation

With the set of point locations $x_i = \{x_1, ..., x_n\}$ and their values $f_i = \{f_1(x_1), ..., f_n(x_n)\}$ we can find function $R(x)$ which gives smooth interpolation of these data using radial basis functions (RBFs) [Carr et al. 2001].

$$R(x) = P(x) + \sum_{i=1}^{n} \lambda_i \phi(|x - x_i|), \quad x \in R^d \tag{4}$$

In equation 4, $P(x)$ is a low-degree polynomial, λ_i is a real valued weigh, ϕ is a basis function, and n is the number of control points. Our RBFs $R(x)$ are defined by the corresponding feature points achieved in section 5.2.1, and its distance vector between the generic hand and source hand. The thin plate spline (TPS), $\phi(r) = r^2 \log r$ is used as our basis function for smooth deformation of our skin mesh, since TPS interpolates specified points while minimizing an approximate curvature [Bookstein 1989].

The vertices of the generic hand mesh are displaced by the RBFs resulting in a clone hand which has the same contour curve, hand geometry, and joint location as the source hand. The depth value of hand geometry was not considered in RBFs deformation, since the features for depth could not be obtained from the 2D source hand image. Instead, the generic hand depth is proportionally scaled by the hand length measured in section 5.1.

6 Results

Our automated method generates a person-specific human hand model with similar joint structure and skin geometry to the source individual's hand. The similarity between the original and cloned hand can be seen in Figure 6(a) and (b). While texturing and animated skin deformation are beyond the scope of this paper, some

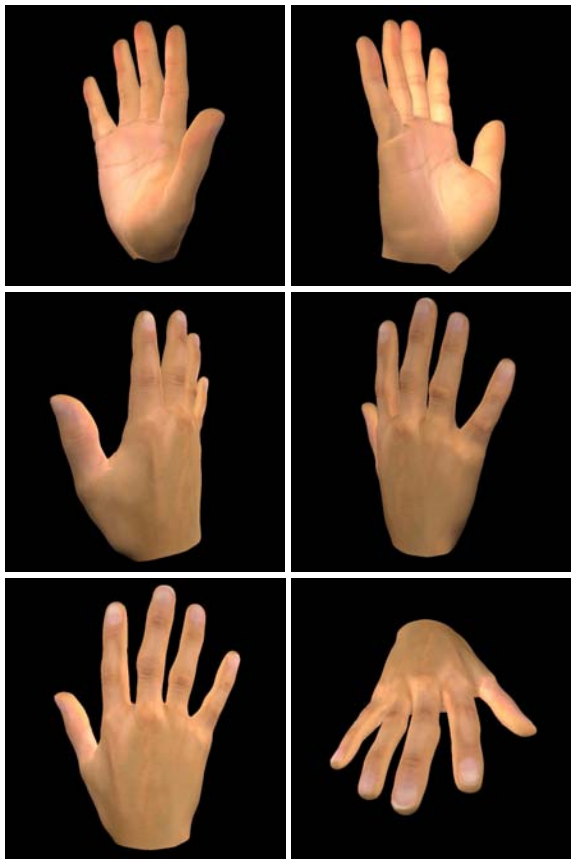

Figure 8: Hand clone result in various view points

supplementary properties of our process help reduce the effort involved in these tasks:

- The source hand image can be re-used as the skin texture map in the 3D clone hand. Due to the similarity of contour shape and scaling between the clone hand and the skin texture, the texture mapping can be executed using simple planar projection in the camera direction.
- Also, because our generic hand model has a joint skeleton bound to skin geometry, it is easy to control the final model using this joint skeleton [Figure 6(f)]. If the generic hand model has animated skin deformation or other properties, the clone hand inherits those characteristics properly.

These supplementary properties are demonstrated in Figure 8, and Figure 1(e) where the individualized model is textured by planar projection, and the skin deformation system from the underlying generic Maya hand model is used to deform the skin guided by rotations of the skeleton joints.

We tested our method by creating five human hand models having different age, sex, and scale; a four year-old girl, a five year-old boy, a large male, an average female, and an average male. We took hand picture in specific conditions described in section 3 and used two generic hand models, a low resolution mesh (13372 triangles with 6715 vertices), and a high resolution mesh (32784 triangles with 16423 vertices). The results are shown in Figure 10 (Color Plate). The first row shows extracted joint information, the second row shows rendered clone model with texture maps, the third row shows the polygon mesh, and each column demonstrates cloning a particular source hand.

Our method requires setting several thresholds to extract creases; Canny edge detection, tensor voting, and crease extraction. Once

set the default thresholds are sufficient during our test provided good lighting conditions (without significant shadows on surfaces) in the source image.

7 Conclusions

In this paper, we presented a method to make a person-specific hand model using the surface anatomy visible in a single photograph of that individual's hand. We extracted basic hand geometry and creases using tensor voting for the surface anatomy features. The joint structure was inferred from the surface anatomy features. Finally, we deformed skin geometry using RBFs with automated feature correspondences based on the curve segment matching.

As a result, we can create a virtual hand clone that has the same contour geometry, joint skeleton, and skin texture map as the source person. Since the algorithm has no human guidance (other than threshold adjustment in difficult cases) and the only information required is a palmar skin image, the method is easily applied.

Since we only use a palm picture, we cannot measure hand thickness accurately, relying instead on an approximate depth proportional to hand length. This approximation may result in an erroneous depth of the joints within the mesh. Because the CMC joint of the thumb has very complex mechanisms in terms of biomechanics and anatomy, accurate CMC joint modeling from a 2D image cannot be guaranteed.

Also, because our method relies on important creases on the palmar skin, we cannot apply our algorithms to an atypical human hand; for example, those with missing or vague important creases, or with corns, calluses, and any significant scar on the palmar surface.

On the other hand, our overall approach does not prohibit the integration of additional information, such as additional texture photographs from other angles, and accurate depth data from a volumetric scanner.

Acknowledgments

We appreciate Pamela Fox, Zhenyao Mo, Genich Kawada, Changki Min, and especially Dr. Tiffany Grunwald in USC medical school for her advice on the anatomy of the human hand.

References

ALBRECHT, I., HABER, J., AND SEIDEL, H. P. 2003. Construction and animation of anatomically based human hand models. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA-03)*, 98–109.

ALIAS. Maya 6. http://www.alias.com.

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2003. The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. Graph. 22*, 3, 587–594.

BOOKSTEIN, F. L. 1989. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE. Trans. Pattern Analysis and Machine Intelligence 11*, 567–585.

BRAND, P. W., AND HOLLISTER, A. M. 1999. *Clinical mechanics of the hand: 3rd Ed.* Mosby.

BUCHHOLZ, B., ARMSTRONG, T., AND GOLDSTEIN, S. 1992. Anthropometric data for describing the kinematics of the human hand. *Ergonomics 35*, 3, 261–73.

BUGBEE, W. D., AND BOTTE, M. J. 1993. Surface anatomy of the hand: The relationships between palmar skin creases and osseous anatomy. *Clinical Orthopaedics and Related Research*, 296, 122–126.

BURT, P., AND ADELSON, E. 1983. A multiresolution spline with application to image mosaics. *ACM Transaction on Graphics 2*, 4 (Oct.).

CARR, J. C., BEATSON, R. K., CHERRIE, J. B., MITCHELL, T. J., FRIGHT, W. R., MCCALLUM, B. C., AND EVANS, T. R. 2001. Reconstruction and representation of 3D objects with radial basis functions. In *SIGGRAPH 2001, Computer Graphics Proceedings*, ACM Press / ACM SIGGRAPH, 67–76.

CURIOUSLAB. Poser 5. http://www.curiouslabs.com.

ELKOURA, G., AND SINGH, K. 2003. Handrix: animating the human hand. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 110–119.

FORSYTH, D., AND PONCE, J. 2002. *Computer Vision A Modern Approach*. Prentice Hall.

GUY, G., AND MEDIONI, G. 1996. Inferring global perceptual contours from local features. *International Journal of Computer Vision 20* (Dec. 06).

HUANG, Z., BOULIC, R., AND THALMANN, D. 1995. A multi-sensor approach for grasping and 3-D interaction. In *Computer Graphics International '95*.

IRIS. Tensor voting framework. http://iris.usc.edu/∼tensorvt.

JAIN, A. K., ROSS, A., AND PANKANTI, S. 1999. A prototype hand geometry-based verification system. In *Proc. of 2nd Int'l Conference on Audio and Video-based Biometric Person Authentication(AVBPA)*, 166 – 171.

KÄHLER, K., HABER, J., YAMAUCHI, H., AND SEIDEL, H.-P. 2002. Head shop: generating animated head models with anatomical structure. In *SCA '02: Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 55–63.

KIM, J., CORDIER, F., AND MAGNENAT-THALMANN, N. 2000. Neural network-based violinist's hand animation. In *Computer Graphics International Proceedings*, IEEE Computer Society, 37–41.

KRY, P. G., JAMES, D. L., AND PAI, D. K. 2002. EigenSkin: Real time large deformation character skinning in hardware. In *Proceedings of the 2002 ACM SIGGRAPH Symposium on Computer Animation (SCA-02)*, 153–160.

KURIHARA, T., AND MIYATA, N. 2004. Modeling deformable human hands from medical images. In *Proceedings of the 2004 ACM SIGGRAPH Symposium on Computer Animation (SCA-04)*, 357–366.

LIN, J., WU, Y., AND HUANG, T. S. 2000. Modeling the constraints of human hand motion. Workshop on Human Motion.

MEDIONI, G., LEE, M., AND TANG, C. 2000. *A Computational Framework for Segmentation and Grouping*. Elsevier.

MOCCOZET, L., AND MAGNENAT-THALMANN, N. 1997. Dirichlet free-form deformations and their application to hand simulation. In *Computer Animation*.

POLLARD, N. S., AND ZORDAN, V. B. 2005. Physically based grasping control from example. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 311–318.

SEO, H., CORDIER, F., AND MAGNENAT-THALMANN, N. 2003. Synthesizing animatable body models with parameterized shape modifications. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 120–125.

SHU, W., AND ZHANG, D. 1998. Automated personal identification by palmprint. *Optical Engineering 37*, 8, 2359–2362.

THOMPSON, D. E., JR., W. L. B., MYERS, L. M., GIURINTANO, D. J., AND III, J. A. B. 1988. A hand biomechanics workstation. *Computer Graphics (SIGGRAPH '88 Proceedings) 22*, 4 (Aug.), 335–343.

TSANG, W., SINGH, K., AND FIUME, E. 2005. Helping hand: an anatomically accurate inverse dynamics solution for unconstrained hand motion. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ACM Press, 319–328.

WU, Y., AND HUANG, T. S. 1999. Human hand modeling, analysis, and animation in the context of HCI. International Conference on Image Processing.

YOUM, Y., GILLESPIE, T. E., FLATT, A. E., AND SPRAQUE, B. L. 1978. Kinematic investigation of normal MCP joint. *Journal of Biomechanics 11*, 109–118.

YU, H. L., CHASE, R. A., AND STRAUCH, B. 2004. *Atlas of Hand Anatomy and Clinical Implications*. Mosby.

ZHOU, K., WANG, X., TONG, Y., DESBRUN, M., GUO, B., AND SHUM, H.-Y. 2005. Texturemontage: Seamless texturing of arbitrary surfaces from multiple images. *ACM Trans. Graph. (Proceedings of ACM SIGGRAPH 2005) 24*, 3, 1148–1155.

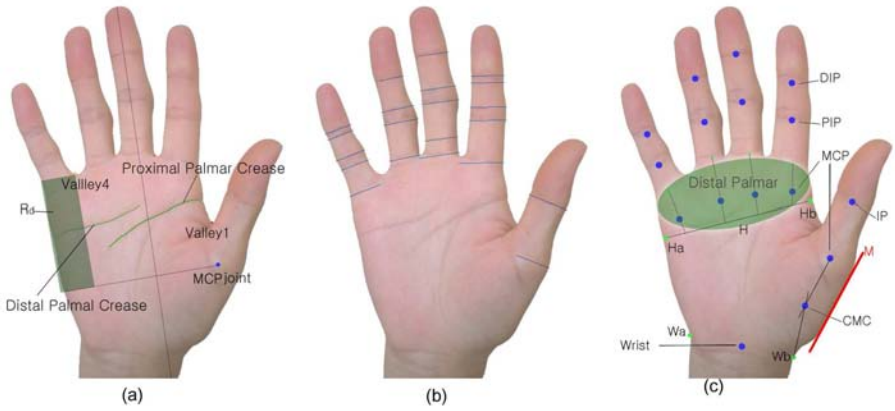

Figure 9: Crease extraction and joint modeling: (a) Green curves are the extracted palm creases (b) Blue lines are the location of finger crease (c) Blue dots are the calculated joint locations

Figure 10: Hand clones of five people: the first row is a set of source hand images and extracted joint information, the second row is a set of rendered clone models with texture maps, and the third row is a set of polygon meshes of the clone models; (a) a four year-old girl of 12cm hand length (low resolution mesh), (b) a five year-old boy of 13cm hand length (low resolution mesh), (c) an average female of 16.5cm hand length (low resolution mesh), (d) an average male of 18.5cm hand length (high resolution mesh), (e) a large male of 21cm hand length (high resolution mesh).