Chapter 1

Introduction

1.1 Automated Human Body Measurement Extraction

The task of human body measurement extraction is to retrieve the body size data from a human subject. The model of the human subject can be built from digital images, and this process can be automated using computer vision algorithms within reasonable processing time.

With the availability of digital cameras and personal computers, digital images have recently become convenient and inexpensive to obtain by average users. Most of the families in the U.S. have computers at home. Most of the laptops currently produced are mounted with integrated cameras. A typical web cam with the ability to capture a resolution (640x480 pixels) well enough for image analysis can cost as few as twenty dollars. Consumers also have become familiar with online shopping since this century. The available technology and the trend in electronic commerce have made it possible for enabling mass production of some custom-made clothes to be ordered online.

1.2 Applications

With the automated human body measurement extraction system, the user will be able to generate their body measurements from home using a PC mounted with a simple digital camera. The system will automatically detect their body measurements and location parameters. Sending the measurement parameters to the online shops will enable custom-made clothes to be ordered through the Internet. The parameters can also be used to produce a 3D model with the shape of an individual's size so that users can simulate trying on different garments. The parameters can also be useful in other applications like aesthetics evaluation

and extraction of key features from individuals. But this research is mostly focusing on the measurement extraction for apparel making.

1.3 Direction of Research

This research focuses on developing a system to automatically take human body measurements based on images with a less restricted background. The system will require the users to provide at least a front image and a side image of themselves. The person will need to keep a specific pose for the picture. The system will estimate the location of the person and the body parts, and then extract the measurements. The application of this system will enable custom-made clothes to be ordered online. Previous studies related to human body measurement have been done with special equipment with complex calibration process and specific environment to process of silhouette extraction. These requirements can be removed with algorithms for image segmentation. Image segmentation algorithms separate the background and the foreground objects in an image with some required user input. The system is seeking a solution that takes human body measurements while minimizing user input. It helps locate the person in the image, guess and provide samples for foreground and background pixels to construct the silhouette, and take measurements on the silhouette.

Chapter 2

Literature Survey

2.1 Body Measurement Extraction Systems

Developing an automated body measurement system has been in research for a long time. In general, there are two major approaches in building a human body measurement system. The first approach involves extracting measurements based on 2D images of a human subject through analyzing the silhouettes. It usually requires at least of a front view image and a side view image of the subject. The second approach involves building a 3D model of the subject by using depth sensors. Then, measurements can be extracted from the constructed 3D model.

2.1.1 Silhouette-based Body Measurement Extraction Systems

In the year 2000, a 2D image-based anthropometric measurement and clothing sizing system was developed by Pierre Meunier and Shi Yinin at the Defense and Civil Institute of Environmental Medicine, in Canada. The system included two cameras in resolution of 1280x960 pixels: one in the front and one on the side. The subject was standing at the center of the two cameras, with a pose holding arms straight down and keeping a distance from the body. Two images were captured simultaneously. A blue backdrop was placed behind the subject to provide a blank background (see figure 1). Calibration markers were embedded on the backdrop to assist the calibration process. The silhouette of the subject was measured based on the two pictures. The measurements were generated by analyzing the silhouette, and taking the stature, the neck circumference, the chest circumference, the waist circumference, the hip circumference and the sleeve length, etc. (See figure 2 and 3.) The experiment

included 349 subjects (95 females and 254 males). Comparing to manual measurements, the system achieved over 99% of reliability. (Reliability was defined as $R = 1 - r^2/s^2$ and r is the measurement error and s is the standard deviation [Meunier & Yin 2000].)

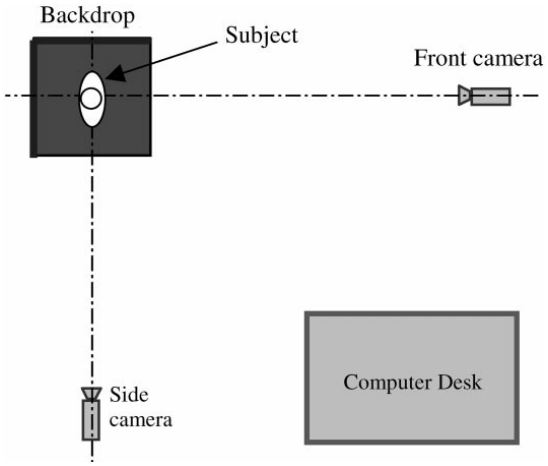

Figure 1: The setup of the 2D image-based anthropometric measurement and clothing sizing system [Meunier & Yin 2000]

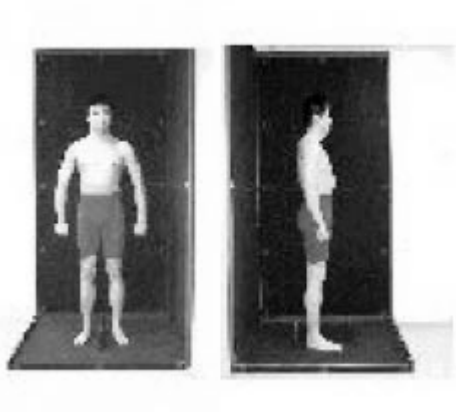

Figure 2: An example of the images taken by Meunier's system [Meunier & Yin 2000]

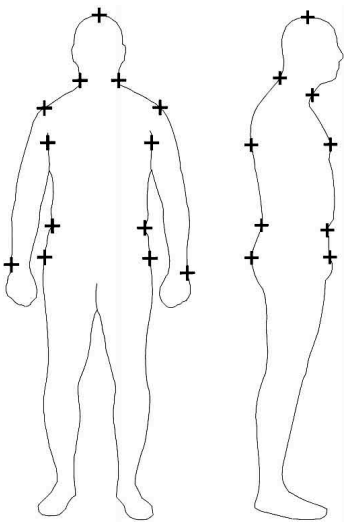

Figure 3: Measurements are extracted based on the silhouette from the frontal image and the side image. [Meunier & Yin 2000]

In 2003, Charlie Wang et al. at the Hong Kong University of Science and Technology designed a clothes-sizing system with three steps: first, from an image, the outline of the person is found using Chan-Vese Algorithm [Virtual Human Modeling from Photographs for Garment Industry, Wang et al. 2003]. Once having the outline, feature points at the extreme ends of the silhouettes are identified, such as hands, armpits, crotch, top of the head on the front silhouette and neck points waist points, hip points on the side silhouette. A template of a standard human 3D model has been constructed ahead of time. This 3D model is reshaped by the given feature points (see figure 4). Then the measurements are calculated from the stretched 3D model [Virtual Human Modeling from Photographs for Garment Industry, Wang et al. 2003]. Using 3D garment feature templates in form of triangle mesh, a clothes try-on can be done on the reshaped 3D human model according the designated feature points [Feature Based 3D Garment Design Through 2D Sketches, Wang et al. 2002].

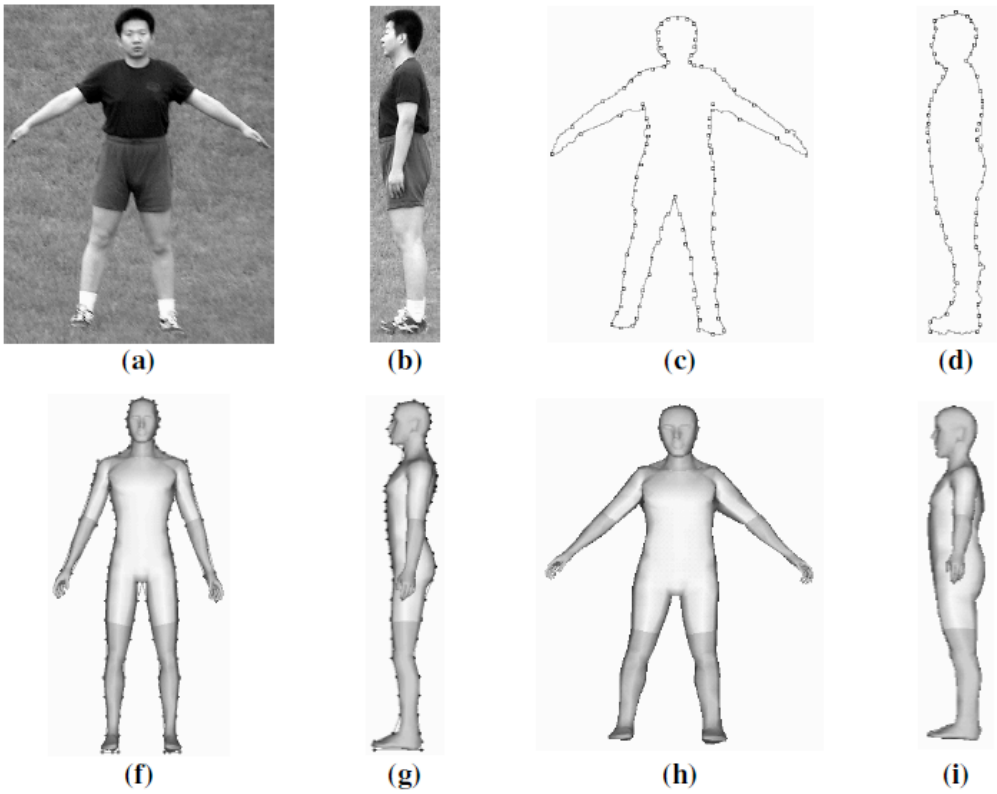

Figure 4: Images a and b are provided by the subject. The images are in a natural background. Images c and d display the extracted silhouettes. Feature points will be measured based on the shilhouettes. Then the standard 3D model will be stretched into the shape of the feature point.. Measurement happens on the 3D model. [Virtual Human Modeling from Photographs for Garment Industry, Wang et al. 2003]

In 2002, Pfister filed a patent on a single camera measurement system. The system goes through four stages: calibration processing, image collection processing, feature detection processing, and tailor measurement processing. In the calibration process, the length per pixel both in the vertical direction and in the horizontal direction are estimated. The calibration process is done by scanning a cross-shaped calibration fixture in front of the camera to locate five points on the calibration fixture. Image collection process will collect four views of the subject: front view, left side view, rear view, and right side view. This system captures the silhouette of a subject without recording its image such that no one can look at any actual photographs. Two silhouettes are needed: front and side. The silhouette is

obtained by using background subtraction with a light colored backdrop to provide a blank background. Once having the silhouette, fuzzy logic will be used to search in a single direction until a large chunk of black lines of pixels are detected. The four searching directions are top-down, left-right, right-left and bottom-up [Pfister 2002].

2.1.2 3D Model-based Human Body Measurement Extraction Systems

In 2006, another 3D body measurement system was developed in Keio University by Koichiro Yamauchi, and Yukio Sata using multiple range images [Yamauchi & Sata 2006]. In this system, a compact range finder was introduced by pairing a digital camera with a laser scanner. Basically, this device could capture images with the sense of depth on each pixel. The system had four poles at each of the corners. The subject stood in the middle. There were four range finders per pole pointing to the subject at the center. There were 16 range finders in total. The system did not require a background but required calibration to establish the world-coordinates at each pole. A 3D model was constructed as a cloud of points (see figure 5). Each point was a pixel from an image plus the distance sensed by the range finder. In the 3D model, hair was usually missing due to the error caused by its specular reflectivity to the laser [Yamauchi & Sata 2006].

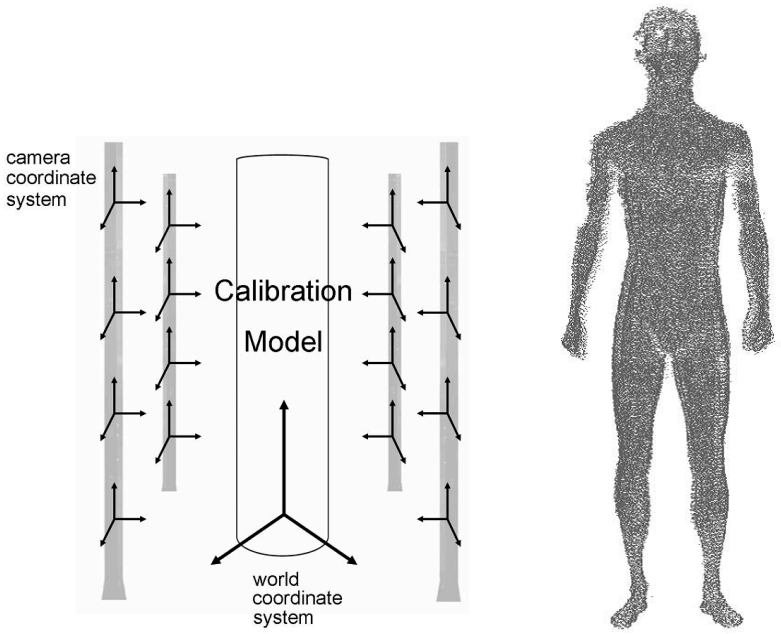

Figure 5: On the left: a collabrated system of 16 range finders is setup to measure the subject in the middle. On the right: a cloud of points are gathered from the range finders to represent the measured subject [Yamauchi & Sata 2006].

Commercial products such as the TC^2 3D body scanner of Textile/Clothing Technology Corporation (TC^2) have been developed and available to the market based on the similar concept of using depth sensors to build a 3D model of a person and extract measurements. An alternative depth sensor employs a projector/camera pair, as appeared in Certain and Stuetzle's automatic body measurement system [Certain & Stuetzle 1999]. The projector throws a pattern of light and dark vertical stripes onto the scene to illuminate the human body. The shapes of the stripes reflected by the body are caught by the camera. 3D models are generated based on the depth cue provided by the shapes of the stripes. Once having the 3D model, the horizontal slices of the 3D human model are analyzed to extract the body measurements.

2.1.2.1 Structured-light 3D Scanner

Many scanners are based on the concept of structured-light 3D scanner. A structured-light 3D scanner is a depth-sensing device that measures a 3D model of an object using a system consisting a pair of a projector and a camera. The projector will project narrow bands of lights as a pattern to recognize. From the distortion of the pattern, the depth cues can be measured to estimate depth value, and retrieve the 3D coordinates on the surface of the object [Structured-light 3D scanner, Wikipedia 2010].

The patterns of parallel stripes are widely used. There are two major methods in stripe pattern generation: laser interference and projection. The laser interference method works with two wide planar laser beams. An equidistant stripe patterns is resulted from the interference of the two laser beams. The size of stripe pattern may be adjusted by varying the angle between the two beams. In the projection method, a slide projector will display a striped film to create the stripe pattern [Structured-light 3D scanner, Wikipedia 2010]. Modern video projectors are capable of displaying digital images, which makes creating of the stripe pattern even easier. According to Certain and Stuetzle [1999], a TC^2 scanner "projects multiple different stripe patterns and uses the time sequence of light and dark recorded by a pixel in the image to find the corresponding point on the slide" [Certain & Stuetzle 1999].

The depth calculation involves analysis of the captured stripe patterns. Stripe width, stripe frequency and phase provide depth cue [Structured-light 3D scanner, Wikipedia 2010]. Using these cues, the phase shifting is calculated to construct an absolute phase map from a set of images. Then a point cloud is constructed based on the phase map [Peng & Gupta 2007].

2.2 Background Estimation Algorithms

Image matting is a technique often used in photography and filmmaking to take a foreground object out of one picture and blend it into a different scene. Notice this process requires the foreground image to be cut out from one image, and then pasted into another scene. To take an object out of an image, it involves estimation of membership of the pixels in an image, as whether a pixel belongs to the background or the foreground. The membership could be in fraction, and understood as transparency. In the other words, it estimates the percentage for a pixel to belong to the foreground so that the extracted foreground object can be smoothly blended to a new scene with this proportion to have a natural look. Image segmentation is a special case with alpha value (the transparency value) restricted to be either 0 or 1 [Zheng et al. 2008].

2.2.1 Chroma Key

One of the most commonly used matting methods employed by TV news is called "bluescreen" [Chroma key, Wikipedia 2010]. The news host stands in front of a blue backdrop. The background is estimated by a color key, where any pixel with a color close to blue is estimated as background pixel, while any pixel with a color other than blue is estimated as foreground. This method is also known as "chroma key". Many modern movies are produced using this technique to blend real actors into a computer-generated scene. This technique was seen in the research of Capo et al. [2006] for capturing silhouettes of a person.

However, chroma key has some apparent limitations: first, the algorithm fails when the foreground object contains some color close to the chrome key. Second, this method must work with the images taken in front of a blank background. It cannot work with any existing image that already has a background.

2.2.2 Background Subtraction

Another technique similar to chroma key is background subtraction. In this technique, two pictures are taken, including an image with the foreground object and an image of the same scene without the foreground object. The background in the two images must be perfectly aligned. A pixel-wised subtraction happens between the two images. Each pixel in the subtraction result with a significant large absolute value is considered caused by the appearance of the foreground object. This technique usually requires the camera to be fixed so that the frames captured through time can be perfectly aligned. This technique was applied in the research of Apostoloff and Fitzgibbon [2004] as the first stage of their system.

In reality, the segmentation result from background subtraction is often very rough. One problem is that the camera may be sensitive and captures a lot of noise. This makes the background pixels vary from frame to frame even without the presence of the foreground pixels. Also, the lighting condition may change over time. The main task in background subtraction is to cope with the instability, as appeared in many forms of research. To cope with the problem of noise and changing of light in the environment, usually a Gaussian statistical model is maintained for each pixel to find the mean intensity and the standard deviation [Matusik 2001]. The pixels with intensity outside the range of three standard deviations to the mean intensity are considered as caused by occurrence of foreground pixels, while the pixels within this three standard deviation range considered as detecting the same background pixel.

Another problem that may cause difficulty in this method is the shadow. Pixel difference may be caused by the blockade of the light source instead of by the blockade of the background by the foreground object directly, which makes the background subtraction method more unreliable and requires additional handling. Since shadows typically cause

change only in pixel intensity, an RGB colored pixel can be transformed into CIE L*a*b color space to measure the distance by excluding the light intensity term (to only consider the chromatic difference) [Elgammal et al. 2000]. Jung [2009] proposed a shadow detection model by considering the intensity of the shadow as a scaled background model with Gaussian noise. A neighborhood on the image is checked in pixel-wise to background model and the rates of change on the pixels are compared. If the pixels with large rate of change are clustered in the neighborhood, a blob of shadow is detected. A more sophisticated model, such as Hidden Markov Model, can be trained on each pixel location to distinguish the appearance of foreground object and the appearance of shadow, as seen in the research of Rittscher et al. [2000].

Another challenge to this technique is that the foreground pixel may match the background pixels, and the segmented foreground region may contain many holes as a result. Also, similar to the chroma key technique, it does not work with an existing image without knowing a well-aligned empty scene.

Besides the color difference, the depth difference can also be used effectively in background subtraction, when employing a depth sensor such as a stereo camera in the research of Crabb et al. [2008] and the research of Gordon et al. [1999].

2.2.3 Image Matting Algorithm

Different to background subtraction, image matting algorithm works with natural environment while not requiring a background image. The background pixels and the foreground pixels are separated based on the user provided sample pixels. Image matting refers to the process of decomposing an observed image I into a foreground object image F and a background image B, with an alpha matte α [Zheng et al. 2008], as

$$I = \alpha * F + (1- \alpha) * B \qquad (1)$$

In this equation, I, F, B and α are 2D grids sharing the same size. Alpha value is the term to define transparency of a pixel. Typically, alpha value is a float value between 0 and 1 inclusively. In this equation, it is apparent that the result image I is an interpolation of the two images F and B. More intuitively, the larger value that α is the heavier of the weight that will be used for the color at the foreground pixel when blended with the new background image.

However, the above is merely a description of alpha blend. The goal of image matting is to estimate the alpha matte α, as the percentage of mixture to blend the foreground pixel from F and the background pixel from B to reach the result pixel in I. Notice that alpha value can also be viewed as the probability for a pixel to belong to the foreground. So, once an alpha matte is obtained for an image I, we could simply apply a threshold (for example, use 0.5 as the threshold) to the alpha matte to obtain a binary silhouette.

2.2.3.1 Types of Matting Methods

The task of matting involves estimating the probability for one pixel to belong to the foreground or background. Many different probability models have been in research. The probability models include Bayesian probabilistic model [Chuang et al. 2001], Poisson equation [Sun et al. 2004], etc. Bayesian matting and Knockout matting are the most popular matting methods, which are used in a lot of research papers as a benchmark for comparing the performance of a new matting method.

Two of the most popular input methods for matting are: matting through a user provided trimap or matting through user provided scribbles [Zheng et al. 2008]. A trimap is an overlaying 2D grid as large as the image, recording each corresponding pixel location as foreground, background, or unknown. The alpha values at unknown pixels are estimated from the user-provided known pixels. On the other hand, the “scribble” method usually requires

the user to give a small sample set of foreground and background pixels by marking on the image.

There exist two other types of input methods for matting that require simpler input. In a more recent matting algorithm "GrabCut", user may provide samples by drawing a bounding rectangle that tightly contains the object [Rother et al. 2004]. The pixels inside or outside this bounding rectangle are considered as the samples for the initial foreground and background. The foreground and background are modified iteratively by minimizing an energy function until convergence. Spectral matting can segment the image without taking user input at all, but is at the risk of being erroneous for images with highly textured backgrounds [Zheng et al. 2008].

Traditional methods like Bayesian matting require a user to provide an initial trimap. This step is not only tedious but also requires a user to provide relevant samples in order to reach a good performance. Automating this trimap initialization step has been in research. A popular approach is to use background subtraction by taking an extra picture of a blank scene to provide a rough initial trimap, as employed in the research of Crabb et al. [2008] and Apostoloff and Fitzgibbon [2004]. Other alternatives seek help from special equipment, like a flashlight in the research of Sun et al. [2006] and a camera that can vary its focal length in the research of Reinhard and Khan [2005] to compare the difference in two frames to provide an initial trimap.

2.2.3.2 Knockout Matting

Knockout matting is one of the early matting algorithms, introduced as a benchmark method in the research of Chuang et al. [2001]. The approach involves finding a single background color and a single foreground color from the nearby samples. It involves the following steps: From an observed pixel C at position x to be estimated, the closest known

foreground pixel sample at distance of d is found. Then, all foreground samples within the distance of $2d$ from position x are collected. For pixels at distance of d, their weights are assigned to be one. For pixels at distance of $2d$, their weights are assigned zero. the weights for pixels between d and $2d$ are linearly interpolated. After the weighted samples have been collected, the foreground color F is calculated as the weighted sum of all these sample colors. Similarly, the background color B can be calculated from the background samples nearby. Then three alpha values are estimated individually on the R/G/B channels separately, where one alpha value is calculated using $(f(C) - f(B)) / (f(F) - f(B))$, where $f(x)$ is a function to read a specific channel of a color. This equation can be interpreted as calculating the ratio of the distance between the observed color and the background color over the distance between the foreground color and background color. Once the three alpha values are estimated, the final alpha value is calculated as a weighted sum of the three alpha values, whose weights are proportional to $|f(F) - f(B)|$. [Chuang et al. 2001]

2.2.3.3 Bayesian Matting

Bayesian matting estimates the alpha value for each pixel with the maximum Bayesian probability based on the background samples and the foreground samples nearby. First, pixels with similar colors are clustered within a local neighborhood. The clustering process can be done by color quantization to reduce the color depth [Chuang et al. 2001]. Then the algorithm will take one pair of clusters at a time: one from the foreground and one from the background. Three values, including a foreground color F, a background color B, and an alpha value α, are estimated at the same time, such that the three values give the maximum Bayesian probability to generate an observed color with the given two clusters of pixels samples. The Bayesian probability equation is as follows: [Chuang et al. 2001]:

$$\arg\max P(F,B,\alpha \mid C)$$
$$= \arg\max P(C \mid F,B,\alpha) * P(F) * P(B) * P(\alpha) / P(C)$$

$$= arg\ max\ L(C|F,B,a) + L(F) + L(B) + L(a) - L(C)$$
$$\approx arg\ max\ L(C|F,B,a) + L(F) + L(B) \qquad (2)$$

The Bayesian equation is a product of conditional probabilities from different sources. By taking logarithmic operation over the Bayesian equation, the equation is transformed from a series of multiplication into a series of summation. The probability terms for $L(\alpha)$ and $L(C)$ are dropped to simplify the model. Then the probability distribution of the foreground and the background pixels is modeled with an elliptical Gaussian probability distribution in the color space, using the mean color and the color covariance [Chuang et al. 2001]:

$$\overline{F} = \frac{1}{W}\sum_{i \in N} w_i F_i$$
$$\Sigma_F = \frac{1}{W}\sum_{i \in N} w_i (F_i - \overline{F})(F_i - \overline{F})^T \quad , \qquad (3)$$
$$L(F) = -(F_i - \overline{F})^T \Sigma_F{}^{-1} (F_i - \overline{F})/2$$

where w_i is the weight distributed by a spatial Gaussian fall-off. The longer distance between C and a pixel color from the foreground results in a significantly smaller weight. The conditional probability is modeled using Gaussian probability distribution [Chuang et al. 2001]:

$$L(C \mid F,B,\alpha) = -\left\| C - \alpha F - (1-\alpha)B \right\|^2 / \sigma^2{}_c \qquad (4)$$

After the two terms defined above have been plugged into the Bayesian equation, a derivative on the probability function equation *arg max P(F,B,α| C)* is taken and the equation is made equal to zero. A set of parameters (F,B,α) that satisfy this equation gives the position in the probability distribution surface that is not increasing, which means the probability is at its local maximum at this position. The equation can be written as follows to solve F and B by treating α as a constant [Chuang et al. 2001]:

$$\begin{bmatrix} \Sigma_F^{-1} + I\alpha^2/\sigma_C^2 & I\alpha(1-\alpha)/\sigma_C^2 \\ I\alpha(1-\alpha)/\sigma_C^2 & \Sigma_B^{-1} + I(1-\alpha)^2/\sigma_C^2 \end{bmatrix} \begin{bmatrix} F \\ B \end{bmatrix} = \begin{bmatrix} \Sigma_F^{-1}\overline{F} + C\alpha/\sigma_C^2 \\ \Sigma_B^{-1}\overline{B} + C(1-\alpha)/\sigma_C^2 \end{bmatrix} \qquad (5)$$

This step is called the "Maximization Step."

After estimating the new *F* and *B* values, the transparency *α* can be estimated using the following equation [Chuang et al. 2001]:

$$\alpha = \frac{(C - B) \cdot (F - B)}{\|F - B\|^2} \qquad (6)$$

This equation treats *α* as an unknown and *F* and *B* as constants. This step is called the "Estimation Step." Then an expectation maximization mechanism is run by applying the new *α* value to the "Maximization Step" again to estimate another round of *F* and *B*. The process repeats until *F, B,* and *α* converge, where the probability reaches a local maximum. The final *α* value is the estimated *α* value for the pixel. Since there can be different pairs of clusters chosen to estimate *α* value at the first step, usually all pairs of clusters are used and the *α* value with the largest probability is taken [Chuang et al. 2001].

2.2.4 Poisson Matting

Poisson Matting is based on the Poisson Equation, where gradient operator is applied to the matting equation $I = \alpha F + (1- \alpha) B$ to find an *α* matte that minimizes the error in the approximated gradient field. After the Laplacian operator is applied, the matting equation becomes [Sun et al. 2004]:

$$\nabla I = (F - B)\nabla\alpha + \alpha\nabla F + (1 - \alpha)\nabla B \qquad (7)$$

The term $\alpha\nabla F + (1 - \alpha)\nabla B$ in the above equation is dropped since it is relatively small respective to $(F - B)\nabla\alpha$. The approximated matte gradient field becomes [Sun et al. 2004]:

$$\nabla\alpha \approx \frac{\nabla I}{F - B} \qquad (8)$$

This means the matte gradient is proportional to the image gradient. For a given image *I*, with a user-provided trimap defining the matte into "definitely foreground," "definitely background" and "unknown," the alpha matte α^* for the unknown region Ω is calculated by minimizing the difference between the term on the left and the term on the right of the equation (8) [Sun et al. 2004]:

$$\alpha^* = \arg\min_{\alpha} \iint_{p\in\Omega} \left\| \nabla\alpha_p - \frac{1}{F_p - B_p} \nabla I_p \right\|^2 dp \qquad (9)$$

In practice, the gradient field ∇I, can be calculated by a gradient operator on the image *I*. The samples of definite background and definite foreground close to an unknown location *p* are used to estimate the term $F_p - B_p$, and equation (8) is further derived to [Sun et al. 2004]:

$$\Delta\alpha = div\left(\frac{\nabla I}{F - B}\right), \qquad (10)$$

where, Δ is the Laplacian operator $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ (as the sum of the second order partial derivative) and *div* is the Divergence operator, where for a vector field $F = Ui + Vj + Wk$, applying the Divergence operator gives $div(F) = \nabla \cdot F = \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z}$ according to Wikipedia [Divergence, Wikipedia 2010]. An iterative optimization is looped to calculate the alpha matte for the unknown region. Then any alpha value in the estimated matte greater than 0.95 or less than 0.05 are assigned to the "definite foreground" or the "definite background" correspondingly. The modified trimap is then used to calculate the alpha matte for the next iteration. The optimization loop continues until the change in the matting results is sufficiently small and no unknown pixel enters "definite foreground" set or "definite background" set [Sun et al. 2004].

2.2.5 Cross-bilateral Filtering Segmentation

2.2.5.1 Strength of Association

The cross-bilateral filter segmentation method is a variant of Bilateral smooth filter. Bilateral filter was developed by Tomasi and Manduchi [1998] as an edge preserving smooth filter, in contrast to Gaussian smooth filter where the edges are blurred after the filtering process. Modified from Bilateral Filter, Cross-bilateral filter considers both of the image and the trimap to perform segmentation as mentioned in the research of Paris and Durand. The logic behind cross-bilateral filter is simple: the closer the two pixels, the larger the likelihood for the two pixels to belong to the same category. The distance is measured both in the spatial space (Euclidean distance between two pixel locations) and in the color domain (Euclidean distance between two colors). Pixels at locations too far away do not have much association to each other. Pixels at locations that are close to each other but differ too much in color do not have much association to each other, either.

The strength of association due to spatial distance is modeled with Gaussian function [Gaussian Function, Wikipedia 2010]:

$$g(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (11)$$

In this equation, x^2+y^2 is the square of the Euclidean distance between the pixel location *(0,0)* and location *(x,y)*. The Gaussian function achieves Normal Distribution around the center at *(0,0)* with *g(x,y)* reaching the maximum value. The value decreases with any enlargement of the distance. σ is the standard deviation of the distribution. Statistically, over 99% of data is covered within 3σ distance [Milton & Arnold 2003, p.118] – thus the width of a bilateral filter is usually chosen as 6σ as having 3σ on each side. For example, for a 2D filter kernel of size *m*-by-*m*, weighted factor is chosen to be $\sigma = m/6$. The magnitude value $1/\sqrt{2\pi\sigma^2}$ is

simply a way to normalize the value so that the summation of all the pixels within the kernel leads to one [Gaussian Function, Wikipedia 2010].

Similarly, the strength of association due to color distance is modeled with another Gaussian distribution. In case of color distance, the Euclidean distance between two colors in the color space replaces the spatial distance term in the previous equation. For example, two colors with *(r1,g1,b1)* and *(r2,g2,b2)* with $\Delta r = r2-r1$, $\Delta g=g1-g2$, $\Delta b=b1-b2$, have a square distance of $\Delta r^2+\Delta g^2+\Delta b^2$ in between.

$$g(r,g,b) = e^{-\frac{\Delta r^2+\Delta g^2+\Delta b^2}{2\sigma^2}} \qquad (12)$$

There is no guideline for choosing a good σ for Gaussian distribution in the color space. It depends on the color model as well as upon the image itself. In general, when a very large σ is in use, the strength of association due to color difference will become constant and have a value close to one. In this case, it means the color distance is not considered at all.

2.2.5.2 Filtering Process

Based on a user provided trimap, the cross-bilateral filter segmentation algorithm goes through a process similar to image convolution: a mask is placed on the top of the image, and moves through the image grid centering at every pixel. Each pixel under the filter mask is compared to the center pixel for both the distance in the spatial domain and the distance in the color space. The strength of association is calculated using the equation mentioned in the previous section. The two values of strength of association are multiplied together. Their product is then multiplied with the value in the trimap. This result is the weight of influence that one pixel in the neighborhood has on the center pixel. The weights from all pixels in the regions are summed into one value, and this value is normalized as the probability for the center pixel to belong to a foreground pixel [Paris & Durand 2006].

The normalization term for each pixel is given by [Paris & Durand 2006]:

$$N = \sum e^{-\frac{1}{2}(\frac{d(\xi,x)}{\sigma_d})^2} e^{-\frac{1}{2}(\frac{d(f(\xi),f(x))}{\sigma_r})^2} \quad (13)$$

The estimated trimap value is given by:

$$T^*(x) = \frac{1}{N} \sum e^{-\frac{1}{2}(\frac{d(\xi,x)}{\sigma_d})^2} e^{-\frac{1}{2}(\frac{d(f(\xi),f(x))}{\sigma_r})^2} T(x) \quad (14)$$

The function *d(x,y)* measures the Euclidian distance between two pixel locations, the function *f(x)* gives the color/intensity read from the image, and function *T(x)* and $T^*(x)$ give the probability before and after the estimation process, variable *x* describes the spatial location of a pixel in the neighborhood and variable ξ describes the spatial location of the center pixel.

2.2.6 Matting Solved as Graph Problem

2.2.6.1 Graph Cut

Graph Cut is another segmentation/matting algorithm taking grayscale or color images as input. It was first proposed by Boykov and Jolly [2001], and then further developed by Boykov and Funka-Lea [2006]. The idea is to treat image segmentation problems as a graph problem and to use the existing algorithms to solve s/t graph by minimizing some energy function to solve a labeling for the given image.

An image is treated as a graph in the following way: each pixel in the image is a node; object label and background label are considered as two terminal nodes; There are edges connecting the neighboring pixel nodes; Every pixel node is connected to the two terminal nodes through two additional edges; Each edge has a weight as the strength of connection. (See Figure 6.) Edges between the pixels are called *n-links* where *n* stands for

"neighbor." Edges between a pixel and a terminal are called *t-links* where *t* stands for "terminal" [Boykov & Funka-Lea 2006].

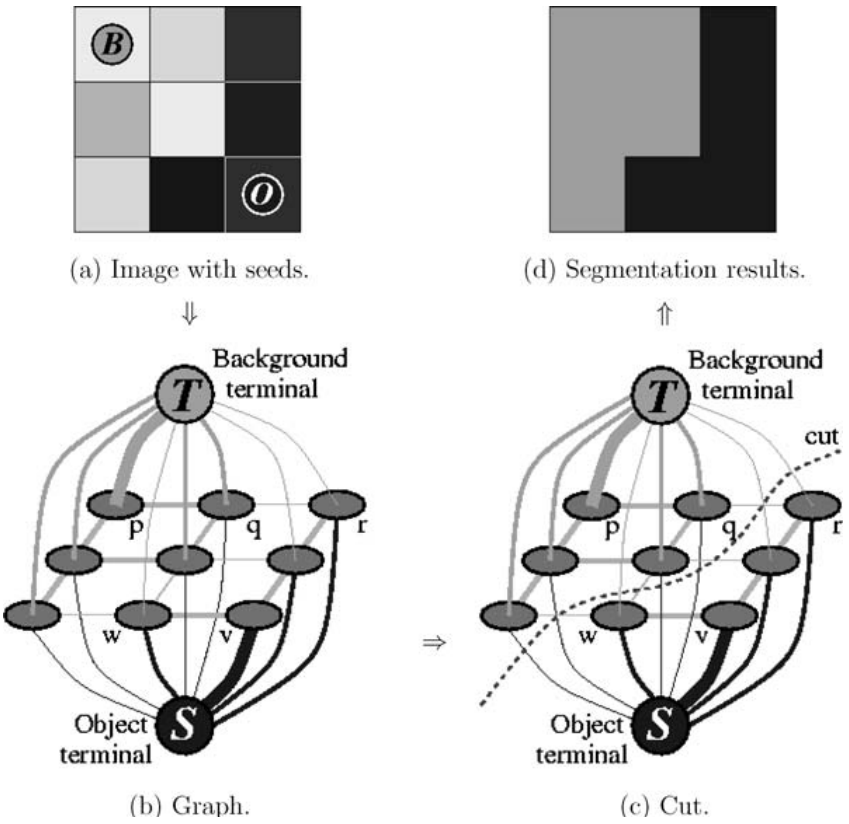

Figure 6: An example of presenting an image segmentation problem into a graph. The *T* and *S* terminals are the "background terminal" and the "object terminal". Sample pixels for object and background are given as seed. Segmentation is achieved by minimizing energy function [Boykov & Funka-Lea 2006].

A standard form of the energy function is [Kologorov & Zabih 2004]:

$$E(f) = \sum_{p \in P} D_p(f_p) + \sum_{p,q \in N} V_{p,q}(f_p, f_q) \qquad (15)$$

In this equation, N is the neighborhood around a center pixel. $D_p(f_p)$ is a function derived from the observed data to measure the cost of assigning the label f_p to the pixel p. $V_{p,q}(f_p, f_q)$ measures the cost of assigning the labels f_p, f_q to the adjacent pixel p,q and is used to impose spatial smoothness. In graph cut, $D_p(f_p)$ is the cost of disconnecting a *t-link* in the graph (as it has to assign a labeling for "object" or "background"). $V_{p,q}(f_p, f_q)$ describes the regional properties of segmentation. And $V_{p,q}(f_p, f_q)$ is the cost of disconnecting an *n-link* in

the graph (as it will happen between an "object" pixel and a "background" pixel) [Kologorov & Zabih 2004].

Boykov and Funka-Lea [2006] modeled $D_p(f_p)$ using the histogram of intensities. For each observed intensity, a probability (or likelihood) of having the observed intensity belong to background / foreground was calculated from the histogram. The cost $D_p(f_p)$ is calculated as negative log-likelihood.

$$\begin{aligned} D_p(\text{"}obj\text{"}) &= -\ln \Pr(I_p \mid \text{"}obj\text{"}) \\ D_p(\text{"}bkg\text{"}) &= -\ln \Pr(I_p \mid \text{"}bkg\text{"}) \end{aligned} \tag{16}$$

$V_{p,q}(f_p, f_q)$ is modeled as a Gaussian function relating to the intensity difference between two neighboring pixels such as:

$$V_{p,q}(f_p, f_q) \propto \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \cdot \frac{1}{dist(p,q)} \tag{17}$$

By using a Gaussian function, the discontinuity between pixels with similar intensity will cause a large penalty. Note that using regional property alone will drive a pixel to always fall into a category according to its color intensity in the global intensity distribution. However, the boundary properties forces spatial coherence to hold within a neighborhood [Kologorov & Zabih 2004].

The total energy will be the sum of the energy on all the edges. Kologorov and Zabih [2004] discovered that the smoothness term $V_{p,q}(f_p, f_q)$ is in the form of Potts model, which is NP-hard to minimize. However, Graph Cut algorithm has been developed to compute a local minimum in a strong sense. A global minimum s-t cut (or close to global minimum) can be efficiently computed in low-order polynomial time adopting a well-known combinatorial optimization as claimed by Boykov and Funka-Lea [2006].

2.2.6.2 "GrabCut"

"GrabCut" is an extension of Graph Cut by iterating the energy minimization step in Graph Cut until the result matte converges to a steady state [Roth et al. 2004]. In Graph cut, a user will mark on the image to provide samples of background pixels and samples of foreground pixels. Then, one round of Graph Cut runs to segment the image based on these user inputs. In "GrabCut", a user will draw a box to enclose the foreground object in the image. Background and foreground Gaussian Mixture Models are constructed by treating the pixels outside the box as the background samples and the pixels inside the box as the foreground samples. Then a min cut algorithm (energy minimization) in Graph Cut is executed to give another alpha matte. This new alpha matte is then be used to construct Gaussian Mixture Models again for the next round of min cut algorithm. The process repeats until the matte does not change. The user may manually edit the result matte to provide better foreground/background samples and run the Grab Cut again, until the user finds result satisfactory [Roth et al. 2004].

2.2.7 FuzzyMatte

FuzzyMatte employes fuzzy logic to measure the strength of fuzzy connection between an unknown pixel from the background and a pixel from the foreground. Based on the strength of connection, the alpha value is estimated [Zheng et al. 2008]. A term called "Affinity" is calculated, which gives the similarity between two adjacent pixels. The Affinity is defined as a weighted sum of the similarity between the pixel and the scribble and the similarity between the pixel and another pixel: [Zheng et al. 2008]

$$\mathrm{A}^{o}(p_1, p_2) = \lambda \mu_{\psi}^{o}(p_1, p_2) + (1-\lambda)\mu_{\phi}^{o}(p_1, p_2), \text{ where } o \in \{f, b\} \quad (20)$$

The pixel-to-pixel similarity is computed using the Gaussian Mixture Model [Zheng et al. 2008]:

$$\mu_{\psi}^{o}(p_1,p_2)=\exp(-\frac{1}{2}\left[I(p_1)-I(p_2)\right]^{T}(\Sigma_{\max}^{o})^{-1}\left[I(p_1)-I(p_2)\right]),o\in\{f,b\} \qquad (21)$$

To calculate the scribble to pixel similarity, intermediate similarity values are calculated using Gaussian Mixture Model by assuming the scribble color as the mean color in the scribble, for both the foreground scribble and the background scribble. Then the pixel to scribble similarity is calculated as a ratio [Zheng et al. 2008]:

$$\begin{aligned}
&\mu_{\phi}^{f}(p_1,p_2)=\begin{cases}1 & if(p_1=p_2)\\ \dfrac{W_{\min}^{f}(p_1,p_2)}{W_{\min}^{f}(p_1,p_2)+W_{\max}^{f}(p_1,p_2)} & if(W_{\min}^{f}(p_1,p_2))\neq 0\\ 0 & otherwise\end{cases}\\
&W_{\min}^{f}(p_1,p_2)=\min[S^{f}(p_1),S^{f}(p_2)] \qquad (22),\\
&W_{\max}^{f}(p_1,p_2)=\max[S^{b}(p_1),S^{b}(p_2)]
\end{aligned}$$

where $S^{f}(p)$ and $S^{b}(p)$ are the similarity metrics derived from the user scribbles, modeled as Gaussian Mixture Model using the mean vector and covariance matrix [Zheng et al. 2008]:

$$\begin{aligned}
S^{f}(p)&=\max_{i}\exp\left(-\frac{1}{2}\left[I(p)-m_i^{f}\right]^{T}(\sum_i^{f})^{-1}\left[I(p)-m_i^{f}\right]\right)\\
S^{b}(p)&=\max_{i}\exp\left(-\frac{1}{2}\left[I(p)-m_i^{b}\right]^{T}(\sum_i^{b})^{-1}\left[I(p)-m_b^{f}\right]\right)
\end{aligned} \qquad (23)$$

The fuzzy connectedness is defined as the strongest connection between two pixels from all the possible paths. The strength of connection is defined as the smallest affinity values between two adjacent points in the path. This min/max metric guarantees the fuzzy connection inside the same region is large, while more importantly, the connection through a smooth color change is large as well. To calculate the fuzzy connectedness, an improved Dijkstra's algorithm using Fibonacci heap is used. The initial iteration is slow, but the modification of scribbles in the following iterations can avoid recalculation of the fuzzy connectedness in a large amount to save computation time [Zheng et al. 2008].

2.2.8 Automatic Matting / Segmentation

The most time-consuming part of the image segmentation/matting is the process of initializing the trimap. Since this step depends on the user to provide, it becomes an uncontrolled factor in the calculation. To overcome this problem and automate the process, various techniques have been introduced to omit the user input process. Many techniques involve analyzing multiple pictures to automatically generation of the initial trimap.

In Crabb *et al.* [2008]'s image segmentation system, a stereo camera is used to separate the foreground object from the background scene through the extra depth information on each pixel. A rough segmentation is obtained from the stereo camera based on the depth result. Foreground objects are the pixels closer to the camera and background objects are the pixels further away from the camera. To fill the imperfection in this initial segmentation, a cross-bilateral filter is applied taking the initial segmentation as an initial trimap.

Reinhard and Khan [2005] employed a camera that can adjust focal length to sense the depth of field. The camera stands at a fixed place and captures a still object in two different focal lengths. In the first image, the camera is adjusted such that the focus is on the foreground object. In the second image, the camera is adjusted to focus on the background scene. Thus, the background scene is blurred in the first image and the foreground object is blurred in the second image. The pixels in the image have their luminance scaled by its log average luminance and applied a sigmoid to make pixel-wise image subtraction more plausible: [Reinhard & Khan 2005]

$$\bar{L} = \exp\left(\frac{1}{N}\sum_{(x,y)}\log(\delta + L(x,y))\right)$$
$$L'(x,y) = \frac{L(x,y)}{L(x,y) + \bar{L}} \tag{24}$$

The result image is then convoluted with two Gaussian blur filters with different kernel sizes. The ratio of the difference in the result from larger Gaussian filter and the result from the smaller Gaussian filter is used as the local variability: [Reinhard & Khan 2005]

$$F(x,y) = \frac{1}{\pi\sigma^2}\exp(-\frac{x^2+y^2}{\sigma^2})$$
$$R_\sigma(x,y) = L'(x,y) \otimes F_\sigma(x,y) \quad (25)$$
$$V_{\sigma 1\sigma 2} = \frac{R_{\sigma 1}(x,y) - R_{\sigma 2}(x,y)}{R_{\sigma 2}(x,y)}$$

The variability in the blurred area gives a very small value, while an area with clear details gives larger variability. Having the two variabilities of the two images, an alpha matte can be calculated by subtracting the variability in the foreground image and the variability in the background image: [Reinhard & Khan 2005]

$$V' = \sum_{i=1}^{n} \frac{\left|V^{fg}_{\sigma_i\sigma_{i+1}}(x,y)\right| - \left|V^{bg}_{\sigma_i\sigma_{i+1}}(x,y)\right|}{\left|V^{fg}_{\sigma_i\sigma_{i+1}}(x,y)\right| + \left|V^{bg}_{\sigma_i\sigma_{i+1}}(x,y)\right| + \varepsilon} \quad (26)$$

In the equation, ε is a small constant added to avoid dividing by zero. The pixels in *V'* that have negative values are the background object, while the pixels that have positive values in *V'* are the foreground object [Reinhard & Khan 2005].

Sun *et al.* [2006] presented a way to generate the initial trimap by using a flashlight and a video camera. The camera keeps capturing frames of the subject, and a flashlight comes during the process. The two adjacent frames with and without the flashlight are taken out from the captured sequence. Since the subject is closer to the camera and the background is further away, the subject will reflect more light and becomes a lot brighter than the way that the background scene reflects the light. An image subtraction is applied to two images and a threshold is applied to get the part of the image that has changed significantly due to reflection of the flashlight. This subtraction result is used as the initial trimap to Bayesian matting, and the alpha matte of the image can be extracted without other user input.

Apostoloff and Fitzgibbon [2004] introduced video matting using learnt image priors. The learnt image priors in their case are simply a background image provided for performing background subtraction. A coarse trimap is generated from the background subtraction technique, followed by a fixed threshold operation to convert it into a binary image and a series of morphological operations (like erosion and dilation) to smooth the borders and fill the holes. Bayesian matting is used as the algorithm to segment the images.

2.3 CIE L* a*b* Color

There is a problem in color interpolation in RGB (red, green, blue) color space because human sensation has some non-linear property when sensing light in different intensities. These properties make the sense of distance between RGB colors uneven. To cope with this problem, a color model called CIE L*a*b* (CIE as "International Commission on Illumination") was introduced to match the non-linear property of the human sensation better. The International Commission on Illumination described all the colors visible to the human eye and create the CIE L*a*b* color to serve as a device-independent model to be used as a reference [3]. Lab color has three channels: L^* is the lightness of a color; a^* is the position of the chrome between red/magenta and green; and b^* is the position of the chrome between yellow and blue. The distance between two CIE L*a*b* color is still calculated by taking the Euclidean distance by treating each color as a point in the three dimensional space. The L^* coordinate ranges from 0 to 100 as the degree of lightness. The possible range of a^* and b^* coordinates depends on the color space that one is converting from. In general, a^* and b^* coordinates fall within a range approximately between 0 and 100.

Transforming between RGB to L*a*b* space involves two steps: first, an RGB color is converted to XYZ color space, given by [CIE 1931 Color Space, Wikipedia 2010]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{b_{21}} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (27)$$

Then the color in XYZ color space is converted to Lab color space [Lab Color Space, Wikipedia 2010]:

$$\begin{aligned} L^* &= 116 f(\frac{Y}{Y_n}) - 16 \\ a^* &= 500 \left[f(\frac{X}{X_n}) - f(\frac{Y}{Y_n}) \right] \\ b^* &= 200 \left[f(\frac{X}{X_n}) - f(\frac{Z}{Z_n}) \right] \end{aligned} \quad (28)$$

, where *Xn, Yn, Zn* is referred as the XYZ color for a white pixel and *f(t)* is non-linear mapping defined as [Lab Color Space, Wikipedia 2010]:

$$f(t) = \begin{cases} t^{1/3} & t > (6/29)^3 \\ \frac{1}{3}(\frac{29}{6})^2 t + \frac{4}{29} & otherwise \end{cases} \quad (29)$$

2.4 Parsing Human Body Silhouette

Recognizing the body parts is an important step before human body measurements can be extracted from the silhouettes/3D models. Many researches, including estimation of human body pose, human body parsing, model-based segmentation and shape decomposition are related to this topic. In general, it is a process to simplify constructed polygon geometry into simpler and more general shapes/features in order to interpret the geometry property or recognize them as conceptualized components. One common approach is to analyze the contour to find the proper points to cut.

2.4.1 Construct Contour

After the segmentation process, the result silhouette is in a form of binary grid. In order to analyze the geometry property of the silhouette, it is necessary to construct its contour first in a form of a chain. In addition to the traditional boundary definition and the extended boundary definition, Ren et al. [2002] described a contour-tracing algorithm constructed as a finite state machine. The algorithm first looks for a start point of a contour; then looks for the next boundary point in the adjacent pixels; lastly terminates when entering a termination state [Ren et al. 2002]. A linear search is conducted and the first pixel that travels from 0 to 1 or 1 to 0 will be considered as the starting point of the tracing. An initial state is assigned to the tracer, depending on whether it is an outer contour tracing or an inner contour tracing. There are 8 states in total, each of which has a predefined sequence to search in the adjacent pixels for the next boundary point. From the current boundary point along with the state, the next boundary point will be looked for in this sequence. After finding the next boundary point, the state of tracer is updated according to the located point in the sequence, and the search starts again at the next boundary point. The trace terminates when entering the state where the contour is closed [Ren et al. 2002].

An alternative approach to construct the contour is to start with a rough contour and modify the contour to obtain a more precise representation through gradient descent method. Morse et al. [2005] proposed a contour construction method using a constraint-based implicit representation. In this approach, the contour is modeled as scattered points in an image, which is initially surrounding the foreground object. A constraint function is defined to give 0 for all points on the curve, and 1 for all known constraints that lie away from the curve. The constraints at other places are interpolated as a sum of weighted effects from the known constraints, where each effect is modeled by a radial basis function. The first derivative of the

constraint function is modeled as a weighted sum of four energy functions, including image energy force, internal energy force, balloon force, and constraint repulsion force. The image energy function makes the points on the active contour to descend into points with some high boundary-like property. The internal energy force reduces the curvature of the active contour. The balloon force decides the direction of gradient descending: shrinking the contour, if the curve is outside the foreground shape, and expanding the contour otherwise. The constraint repulsion force encourages uniform distribution of the constraints along a contour, so that the points will not descent in a clustered fashion. The first three functions follow the direction of the normal of the curve and the 4th function follows the tangent direction of the curve [Morse et al. 2005].

2.4.2 Minima Rule for Visual Salience Detection

In 1997, Hoffman and Singh proposed a contour segmentation algorithm based on the psychological impression of human vision. To find a visual salient point, they defined a minima rule: all points with negative minima of curvature of its bounding curve are boundaries between parts (see figure 9), assuming the minima points on the bounding curve of the silhouette are not influenced due to noise and other sources. Two concepts are defined in calculating the minima rule: the curve normal vector and the curvature vector. The curve normal is perpendicular to the curve at a given point on the contour. The direction of the curve normal is defined to be always pointing inside to the silhouette. The "curvature" is calculated as the rate of change in the normal direction along the curve. The local minimum point having a negative curvature value is considered as a point to cut [Hoffman & Singh 1997].

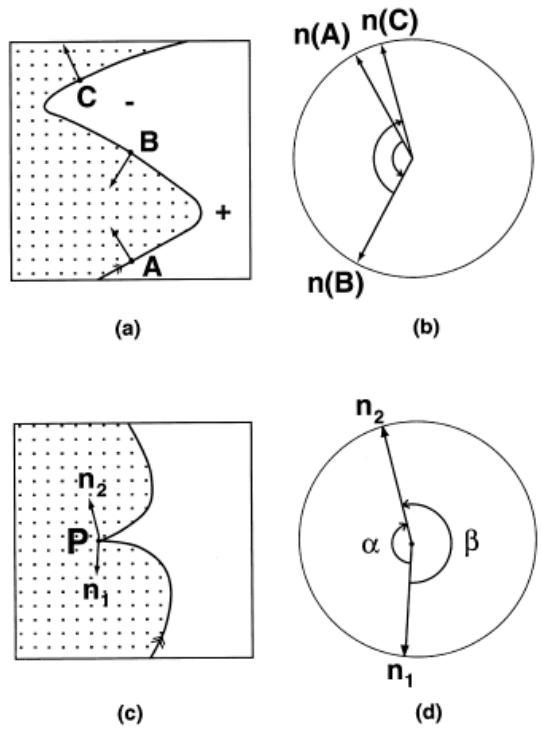

Figure 7: The negative minima curvature points are defined as the points with a negative change in curve normal. Notice every valley on the curve will have a negative change in direction of normal vectors (meaning turning by an angle greater than 180 degrees). The shaded area describes the object and the non-shaded part describes the background [Hoffman & Singh 1997].

In practice, the direction of the curve is calculated as the first derivative; the normal is calculated as the second derivative. Again, the "curvature" here is calculated as the change in the direction of normal, which can be calculated as the dot product of two adjacent normal vectors. (Notice conventionally curvature is calculated as the second derivative. The "curvature" here has a different definition.)

Negative minima of curvature is widely used in many researches/applications in analyzing silhouette. For example, in the automatic human body modeling and motion captured system developed by Capo et al. [2006], negative minima of curvature is applied for estimating the pose of the upper body, including the position of head, elbow, shoulder, and hand. Negative minima of curvature is quite often used as a starting point for improving some silhouette analysis algorithm or performance comparison, as seen in the research of Mittal et

al. [2003] and the research of Latecki and LakaEmper [1999]. Though effective in some cases, negative minimum curvature is not robust when the image is noisy due to its nature of local region analysis [Wang 2004].

2.4.3 Short-cut Rule

The short-cut rule divides silhouettes into parts using the shortest possible cut (See figure 7). A cut is described as a straight line, going across the axis of local symmetry between two points on the outline of a silhouette, such that at least one of the two points has negative curvature. Human vision prefers the short-cut rule according to the research of Singh et al. [1999].

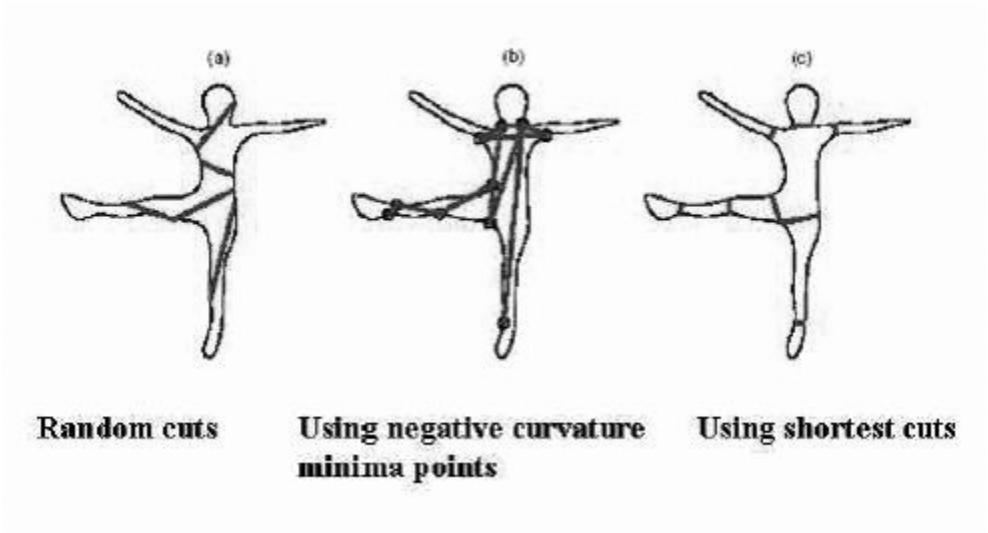

Figure 8: Comparing the results from different cutting strategies. Shortest cut gives the most comfortable visual segmentation to meet human perception [Singh et al. 1999].

Mittal et al. [2003] built an algorithm in estimating the human body pose. First, a human shaped contour is constructed using Bayesian matting on each pixel. Then a combination of short-cut rule and salience requirement is applied to constrain the end point detection for cuts. It starts by locating a point P with negative minima of curvature. Then, it finds another point P_m to pair with P such that the two points divide the contour into two

curves C_l and C_r of equal arc length. A short-cut point P_l on C_l is found by minimizing the equation:

$$P_l = \arg\min_{P'} \left|\overline{PP'}\right|, s.t. \frac{\left|\widehat{PP'}\right|}{\left|\overline{PP'}\right|} > T_p, P' \in C_l, \left|\overline{PP'}\right| \in S \qquad (30)$$

This equation shows that to locate the cut P_l on C_l by finding the smallest distance of PP_l, the arc length of the PP_l must be greater than the distance of PP_l for a threshold value, such that P_l will not come from its immediate neighbors (See figure 8).

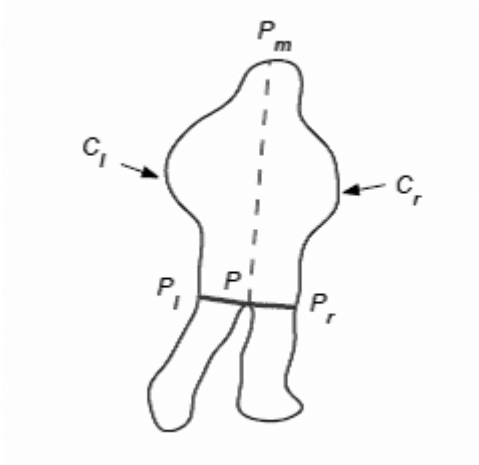

Figure 9: The short-cut points P_l and P_r are located by the cut that has the shortest distance to the crotch point to successfully divide the legs from the silhouette [Mittal *et al.* 2003]

Similarly the other cut point on C_r can be located. The silhouette is then divided into regions by the cuts. To classify a region as a certain body part, multiple assemblies of the regions are analyzed. The likelihood for assembling regions into a human model is estimated with combinations of assigning each region a different body part membership. The combination of body part assignment with the largest probability gives the correct parse of the human silhouette [Mittal et al. 2003].

2.4.4 Maximal Convex Arc in Shape Decomposition

In 1999, Latecki and Lakamper followed another approach in decomposing a shape. To decompose a shape, two consecutive line segments from the original shape were combined into one line segment by eliminating the common vertex and connecting the two other ends. If the two consecutive line segments were chosen correctly, the overall visual impression of the shape should change in the minima degree. The theory was based on the observation that visual parts were in nearly convex shapes [Latecki & LakaEmper 1999]. The Maximal Convex Arc stated that the longest convex curves, also called "supported arcs", should be reserved to keep the original overall shape while eliminating the smaller bumps on the contour. To decide the contribution of each pair of line segments to the maximal convex arc, the cost of the pair of line segments in the contour was calculated using the following formula: [Latecki & LakaEmper 1999]

$$K(s_1, s_2) = \frac{\beta(s_1, s_2) l(s_1) l(s_2)}{l(s_1) + l(s_2)} \tag{31}$$

The variables s_1 and s_2 represent two consecutive line segments on the outer boundary of the silhouette. $\beta(s_1, s_2)$ is the angle of turning between the two line segments. It is kept as always positive. The function $l(s_1)$ gives the length of the line segment s_1, and similarly for $I(s_2)$. Basically, the formula states that lines large both in length and turning angle have a higher priority to be reserved than lines that are either flat or short. Thus, the pairs of line segments with smaller costs should be eliminated first. The process goes on in a loop to "evolve" the silhouette into a more generalized shape (see figure 10). The reserved vertices in the end result are the ones that visually define the object.

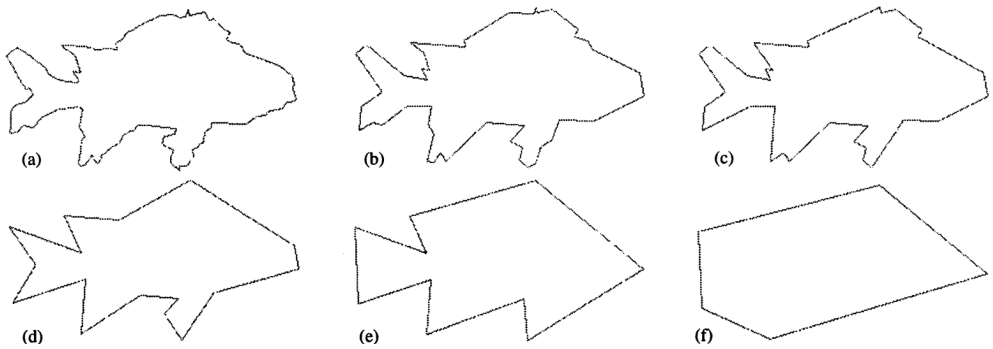

Figure 10: Combining two consecutive line segments with the least cost to preserve the overall shape of the silhouette [Latecki & LakaEmper 1999].

This method is translation invariant, rotation invariant, shift invariant, reflection invariant and scale invariant (If scaling *x* and *y* direction at the same proportion) [Latecki & LakaEmper 1999].

2.4.5 Locating Key Points on 3D Model

When analyzing a 3D model of human body, the 3D model is often intersected with a plane to produce a 2D slice to be analyzed. Unlike 2D silhouettes, 3D models may have different slices when intersected by a plane. In Zhong and Xu [2006]'s proposed algorithm, a set of algorithms to locate key points for measurements on a 3D human model is proposed. First, the crotch point is located as the front silhouette projected from the 3D model. Two methods are proposed to locate the crotch point. In the first method, the algorithm assumes that the ends of the two thighs are not contacted. Thus, a scanning on the horizontal slices for the minimum distance between the pair of points each from one leg is conducted. In the second method, the algorithm handles the case of contacted thighs. In the horizontal slices, the two thighs are two detached ellipses. The ellipses contact and form into a single contour at the position where thighs contact. The middle part of the outline in the front of the horizontal slice is scanned. When two thighs are joined without the torso in the slice, there will be exactly one high peak in this region of the slice. However, once torso enters the

horizontal slice, multiple high peaks start to exist. The location having multiple peaks is detected is the location of the crotch [Zhong & Xu 2006]. To locate the shoulder point, a reference point is roughly placed at the *y* location of the neck and at the *x* location of the bounding box containing the 3D model. Then, the point around the shoulder region with the closest Euclidean distance to the reference point is considered as the shoulder point. The armpit is located using "Minimum inclination angle" between two neighboring triangles (that shares the same edge) around the region of shoulder in the 3D model [Zhong & Xu 2006]. "Minimum inclination angle" refers as the degree of folding between the planes of two neighboring triangles.

2.5 Definition of the Body Measurements

The measurements used in the apparel industry have their own definitions. Chest (or bust for women) is defined as the part of the body "between the neck and the abdomen, enclosed by the ribs and the breastbone" [Lee 1994, p.7]. The chest girth is measured as "the maximum horizontal girth at (chest) bust level measured under the armpits and across the nipples with the subject breathing normally" [Lee 1994, p.9]. During measurement, it is important to let the tape to pass over the shoulder blades to obtain the fullest point of the chest, as defined by Aldrich [2008, p.178]. To reach an easy fit of the clothes, two inches are added to the chest measurement [Tuit 1974, p.9].

Neck is "the part of the body linking the head and trunk", according to Lee [1994, p.7]. The neck girth is measured "below the Adam's apple and at the level of the 7^{th} cervical vertebra" [Lee 1994, p.9]. For apparel purpose, neck base girth is usually measured as the round base of neck, around the neck passing over the base on the 7^{th} cervical vertebra, the intersection points of shoulder and neck, and the upper border of the collarbone at the front, as defined by Lee [1994, p.9] and Aldrich [2008, p.179].

Shoulder is “the joint that connects the arm and the trunk” [Lee 1994, p.7]. Shoulder width is usually measured from the intersection of the shoulder and neck, along the shoulder, to the end of the shoulder bone, with subject’s arms hanging naturally [Aldrich 1997, p.140].

There are several definitions of sleeve length, depending on what kind of clothes the sleeve is made for. Sleeve length for one-piece sleeve is measured from shoulder bone to wrist bone [Aldrich 1997, p.140]. Sleeve length for two-piece sleeve is measured with the arm raised and bent, from center back (15 cm below neck bone) across back; continuing through at elbow point, down sleeve to wrist bone [Aldrich 1997, p.140]. Sleeve length for shirt is also measured with the arm raised and bent, “from neck bone across shoulder to elbow point, then through elbow to wrist bone” [Aldrich 1997, p.140]. Alternative sleeve length can be obtained from the locations with the arm straight. To achieve ease of fit, an extra 0.5 inches can be added to the measurement [Tuit 1974, p.9].

Sleeve inseam, which is related to arm-length, is defined “the distance measured from the right shoulder point, along the outside of the arm, over the elbow, to the far end of the prominent wrist bone, with the subject’s right fist clenched and placed on the hip and with the arm bent at right angle” [Lee 1994, p.19].

Wrist is simply “the joint that articulates between the end of the lower arm and the hand” [Lee 1994, p.8].

Waist is the part of the body at “the location between the lowest rib and hip identified by bending the body to the side” [Lee 1994, p.8]. Waist girth is measured at “waist level with the subject breathing normally and the abdomen relaxed” [Lee 1994, p.9]. Trouser waist position is measured “4cm below the natural waist” [Aldrich 1997, p.140]. This is a low waist measurement and is the common position in which trousers are worn [Aldrich 1997, p.140]. To achieve ease of fit, an extra inch can be added to the measurement [Tuit 1974, p.9].

High-Hip is at “the level of the fullest part of the abdomen that is located approximately 7.5 centimeters or 3 inches below the waist and parallel to the floor” [Lee

1994, p.7]. High hip girth is measured as “the horizontal girth of the high hip measured at the level of the high hip with the subject’s abdomen relaxed” [Lee 1994, p.13]. By heuristic, high hip is approximately 4” below the waist [Tuit 1974, p.6]. To achieve ease of fit, two inches can be added to the measurement [Tuit 1974, p.9].

Hip is “the outer bony prominence of the upper end of the thighbone” [Lee 1994, p.7]. Hip girth is sometime measured with the subject sitting, at the fullest part of hip [Aldrich 1997, p.140]. By heuristic, the widest (or fullest) part of the hips may be “anywhere between 7” and 11” below the waist” [Tuit 1974, p.6]. Or the fullest part can be estimated as the location 21 cm below the waistline according to Aldrich [2008, p.178]. To achieve ease of fit, two inches can be added to the measurement [Tuit 1974, p.9].

Crotch is “the body area adjacent to the vertex of the included angle between the legs, or the level of the base of the buttocks” [Lee 1994, p.7]. Crotch-height is measured as the vertical distance measured from “the midway point of the crotch to the soles of feet” [Lee 1994, p.16]. This vertical distance from the crotch to the ground is measured as the pant inseam, or inside leg-length as shown by Aldrich [1997, p.140].

Chapter 3

Research Goal

3.1 Objectives

The overall goal of this research is to create a software system that extracts human body measurements using images taken from a single digital camera (a web cam on a desktop PC/laptop) without assistance from special calibration equipment. The detailed objectives are as follows:

- To design and implement a silhouette extraction algorithm using a PC/laptop mounted with a web cam (or a digital camera on a stand to take the images of a person).
- To design and implement a silhouette extraction algorithm to extract a human silhouette from the captured image as a closed shape without holes and without background and noise.
- To design and implement a measurement extraction algorithm to extract the critical nine parameters for designing clothes (see methodology for details) are extracted from the silhouette of each subject.
- To collect silhouettes of 30 subjects (in order to reach statistical relevance) conduct measurement extraction, and calculate the average error rates.
- To collect 3D models of 30 subjects (in order to reach statistical relevance) using TC^2 3D body scanner.
- To compare the accuracy of the new method to the existing human body measurement system:
    - The results from this body measurement system with the actual body measurement will be compared for error rate.

- The error rate of this system will be compared with the error rate of the existing body measurement systems such as TC^2 3D body scanner.

- To develop a Graphical User Interface based application for this body measurement system.

Chapter 4

Methodology

4.1 Silhouette Extraction and Measuring

There are two steps involved in the system: silhouette extraction and silhouette measuring.

Background subtraction is simply subtracting two images (with/without the person) to roughly estimate the subject in the image. (For details about background subtraction, refer to literature survey on background subtraction in section 2.2.2.) To perform background subtraction correctly, the environment requires the light to come from the above of the test subject to avoid the blockade of the light sources. But the light source should not be directly on top of the subject because it will cause a shadow at the feet of the test subject. Initially, the test subject should move around to find a place without shadows at the feet to be captured by the camera. The background should be as blank as possible (low textured) in a bright color. (A white background is preferred.) A Logitech web cam in resolution of 640x480 pixels was used in this research.

In silhouette extraction, two pictures of the subject are taken by the system: one front view image and one side view image (see figure 11). First, the camera will be calibrated to construct a background statistical model without the subject being present by taking a series of background images. Then, the subject will be photographed with a predetermined pose. (See figure 11.) When posing for the front image, the subject needs to lower its arms, keep its arms straight, and keep them apart from the body at a distance so that the armpits are not covered and cleanly visible. Also, the subject will need to keep a gap between the two feet, such that the crotch point can be shown without being covered by the two thighs. The subject must stand at the same location when taking the two pictures. The program employs a

segmentation algorithm to filter away the background and the foreground. The segmentation algorithm is a combination of background subtraction and cross-bilateral filtering. (The details of the segmentation algorithm are discussed in section 4.2.)

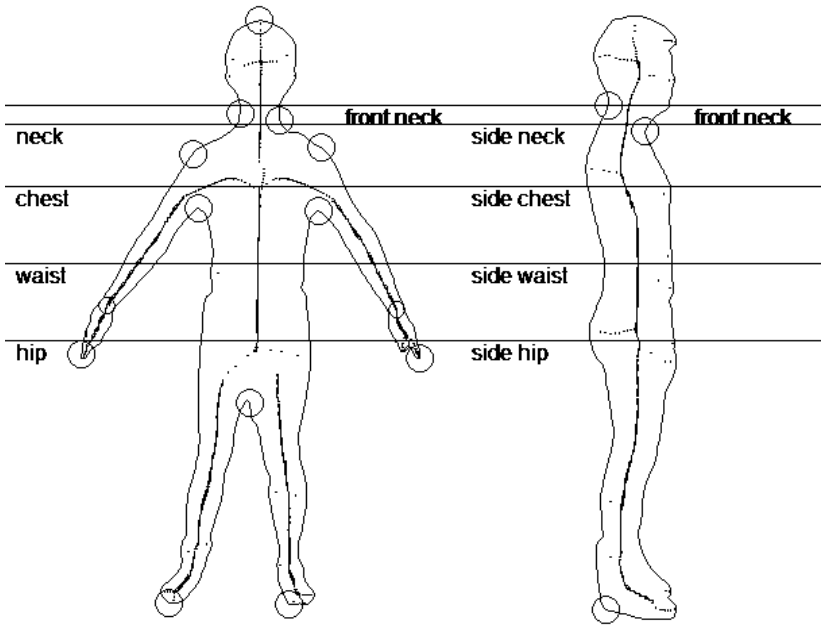

Figure 11: The front view and the side view silhouette of one subject. Key points are located as circles on the contour. Skeleton is used to guide the detection of the wrist. The locations of the body parts are cross-referenced between the two views.

In the silhouette measurement, horizontal and vertical projections are performed on the binary silhouette. The method of calculating the parse of the contour to reach the largest consistency is used to locate many key points on the body. Skeleton is taken to help estimate the narrowest location on the arm. (For details, see section 4.4 “Measurement Heuristics.”)

4.2 Automated Image Segmentation

Background subtraction and cross-bilateral filtering both have their pros and cons in image segmentation. Background subtraction is fast to operate and depends on little user input but may result in noise on background and holes on foreground in the silhouette. These problems can be moderated using a blank background and stabilizing the light sources to

avoid holes and noises but the defects cannot be completely eliminated. On the other hand, cross-bilateral filtering is more accurate but requires a lot of user input to initialize the trimap. The quality of the silhouette depends on the relevance of the initial trimap that the user provides as mentioned. A combination of the two techniques can create a silhouette extraction system that can both require little user input and obtain accurate results at the same time. This initial trimap generation approach is inspired from the research of Crabb et al. [2008] and Apostoloff & Fitzgibbon [2004], as mentioned in section 2.2.3.1. In our new system, a regular camera is used instead of a stereo camera along with the cross-bilateral filter method.

The cross-bilateral method is developed as a variant of bilateral filter (for details, see section "Cross-bilateral filtering segmentation." In general, the algorithm will process within a local filter region, comparing the difference between the color in the center pixel and the colors of the adjacent pixels. Depending on how "close" the two pixels are to each other, the confidence value about whether the pixel belongs to the foreground will be given. The closeness of two pixels is defined both in pixel spatial distance and color distance in L*a*b* color space. The spatial distance is the Euclidean distance of two pixels' locations component *(x,y)* in the image, while the color space distance is the Euclidean distance of two pixels' L*a*b component *(L,a,b)*. The strength of association due to distance is transformed with Gaussian distribution function.

In the proposed system, the background subtraction technique will first be applied. The camera needs to be calibrated with several consecutive frames of an empty background. The frames from the first few seconds (usually five seconds) are disregarded. This is because many cameras available today automatically adjust their sensors to adapt to the environment, which is not desirable in our system. Ideally, the functions of the cameras that automatically balance the colors and white intensity should be turned off. This is because background subtraction algorithm favors a stable background and automatic adjustment causes changes

without any blockade. The skipping of frames is an attempt to accommodate the camera with these undesired self-calibration properties. After the camera becomes stable, twenty consecutive frames are taken from the camera. The images are converted into CIE Lab color space. The mean and the standard deviation at each pixel are calculated based on these frames. The mean of the empty frames will be constructed to an empty background to be used for background subtraction. When the subject poses, the camera will constantly take pictures and convert them to CIE Lab color space at the same time. The subtraction between the constructed empty background and the scene with the subject gives a rough silhouette of the subject. A threshold operation is taken on the subtracted result to make it into a binary image. An initial trimap will be guessed from the rough silhouette. The white pixels will be considered all as foreground pixels. The black pixels that are too far away from any white pixel are considered as background pixels. (The term "too far away from any white pixel" is defined as having no white pixels around it within a box of 15x15 pixels). The black pixels that are not "too far away from any white pixel" are taken as unknown pixels to be re-evaluated. The cross-bilateral filter will have a spatial filter size greater than the "too far away distance." The filter size is usually taken as a box of 51x51 pixels so that for every unknown pixels, several foreground samples can be caught around it. Foreground pixels are assigned one in the trimap, while background pixels and unknown pixels are assigned zero and 0.5 respectively. This initial guess of the trimap is fed as the input to cross-bilateral filter. The segmentation after the cross-bilateral filter gives a more precise silhouette.

There is a conflict of interests in the initial trimap generation, including more black pixels as unknowns can fill holes inside the silhouette better while including less black pixels as unknowns can receive a finer segmentation at the border of the silhouette. Usually it is easier to allow more holes because they are easier to detect and filter at the post-processing step. (See section 4.3 "Silhouette Post Processing.")

4.2.1 Algorithm Optimization Techniques

In the step of trimap generation, the process for detecting a nearby white pixel is slow following the brute force approach. This process can be accelerated with summed-area table – a technique that allows retrieval of the sum of number within a rectangular region in a grid with O(1) time. Summed-area table is an idea of dynamic programming. It involves the following steps: First, calculate the sum of the values under all possible rectangles from location *(0,0)* to location *(x,y)*. This can be constructed using a recursive rule based on the result generated in the previous round by [Crow 1984, Viola & Jones 2002]:

$$sat(x, y) = i(x, y) + sat(x-1, y) + sat(x, y-1) - sat(x-1, y-1) \qquad (32)$$

, where *sat(x,y)* gives a read of the summed-area within the box *(0,0,x,y)*, and *i (x,y)* is the value at the *(x,y)* location in the grid. This can be constructed in O*(w*h)* time. Once the summed-area table is calculated, the sum of the values within the box can be retrieved with this equation [Crow 1984, Viola & Jones 2002]:

$$sum(x_1, y_1, x_2, y_2) = sat(x_2, y_2) + sat(x_1, y_1) - sat(x_1, y_2) - sat(x_2, y_1) \qquad (33)$$

The function *sum(x1,y1,x2,y2)* gives the sum within the box defined by *(x1,y1,x2,y2)*. This can be done in O(1) time.

4.3 Silhouette Post Processing

After a silhouette is extracted from matting, the border may be rough and zigzagged (see figure 12). This rough border can cause difficulty in contour construction and inaccuracy in measurements. To smooth the silhouette, a 5x5 median filter is applied to the binary silhouette, (which is effectively a major filter). Small holes are filled by the median filter too. Then, all small blobs in the background or foreground need to be eliminated using flood fill. The final result silhouette contains exactly one smooth foreground blob without any hole.

Figure 12: left: the original image directly from silhouette extraction. The image does not have a smooth border and may contain holes. Right: the silhouette after a major filter is applied.

The algorithm of flood fill simply says: from one white pixel, get all other white pixels that are connected to this pixel (directly or indirectly). This can be easily done using a queue to include seeds and a image buffer to hold the filled pixels. Initially, the first given pixel is inserted into the queue as an initial seed. Then, the queue is populated by including the white pixels adjacent to any seed in the queue as new seeds. Once a pixel is put as a seed, it needs to be erased from source image (so that it would not be picked again), and the corresponding pixel should be filled in the destination image (to record the blob). Once a seed has been used to generate new seeds, the seed will be removed from the queue. The process continues until no seed is left in the queue. The end target image is the blob. Notice that the number of pixels could be easily counted along the way of flood fill. In this way, we could always keep the blob with the largest size and fill any blob in a smaller size.

Directly performing a flood fill operation will remove all the smaller foreground noise. Inverting the image and performing a flood fill operation will fill all the smaller background holes.

4.4 Measurement Heuristics

The measuring methods employed in this research consist mostly of simple heuristics that take advantage of the priori knowledge about the human body silhouettes and the pose. The key locations on the extracted body silhouette are estimated using these heuristics. Measurements are then conducted based on these key locations.

4.4.1 Body Parts Location by Using Max/Min

With the method of looking for local max or local min, the subject is analyzed to find the neck, the chest, the waist, and the hip location.

First, a horizontal projection and a vertical projection are applied to the front image and the side image. Horizontal projection is simply counting the number of white pixels in a row across the silhouette. Vertical projection is simply counting the number of white pixels in a column across the silhouette. However, in practice this evaluation is not enough because there could be disconnected line segments on the same row due to overlapping of different body parts. For example, at the location of the waist, horizontal projection captures both pixels caused by the body and the pixels caused by the arm. (See figure 11.) To cope with this problem, the longest continuous horizontal line segment in a row is taken instead as the horizontal projection to skip the pixels from the arm.

The location of the neck, the chest, and the hip can be estimated from the side view silhouette, as the narrowest or the broadest parts in the different search regions of the horizontal projection. A linear search is used to locate the local maximum and the local minimum at a specified region in the projection. To locate the chest, a region from the top within 1/3 of the body length is searched for a local maximum. To locate waist, the region from the 1/3 of the body length to the 3/5 of the body length is searched for a local minimum. To locate hip, the region from middle point to the 3/4 of the body length is researched for a local maximum. To locate the neck, the region from 1/14 of the body length to the 1/3 of the body length is search for a local minimum. The waist location is estimated between he chest and the hip, as the location having the smallest circumference. (The circumference estimation is discussed in section 4.4.4 “Ellipse Model in Measurement”.)

The vertical central axis in the front silhouette is also calculated because it is useful in locating the key points. By definition, the vertical central axis is a vertical line that evenly divides the silhouette. To find this line, a vertical projection is made on the front silhouette.

The total number of pixels is counted by summing up the values in the vertical projection. A linear search is applied from left to right to try all possible vertical dividing lines, until the number of pixels falling on the left of the dividing line is more than half of the total number of pixels. This location is the vertical central axis.

4.4.2 Body Parts Location by Contour Tracing

In order to correctly detect key points from contour, the subject needs to pose with arm dropped and kept away from the body in the front view image. If done correctly, the armpits should be revealed on the border of the silhouette. The subject needs to separate its feet far enough such that the crotch point can be captured on the border the silhouette. The silhouette image is first converted into an edge image using Canny edge filter. Canny edge filter and contour construction algorithm can be implemented following the papers of Green [2002] and Chang et al. [2004]. Since edge detection and contour construction are not the focus of this study, the implementation of Canny edge filter is directly taken from OpenCV in this study. Then, an OpenCV implementation of the contour functionality “FindContours” is used to extract contour from binary silhouette. In this experiment, the silhouette is expected to have exactly one contour. To avoid OpenCV from extracting multiple contours from a closed silhouette, a 5x5 median filter is applied to make correction of small loops caused by rough border, which works in most cases. Also, the silhouette should not touch the border of the image. Otherwise, OpenCV may fail to extract contour at all, or extracting a contour without enclosing all the white pixels.

Two types of key points are located on the contour: the border points and the turning points. At first, we need to find the border points: the top of the head, the fingertips of both hands, and the tips of both feet. Since the test subject is always posing in a certain way, these points are easy to find by locating the pixels at the boundary of the bounding rectangle. This

is done by finding the pixels on the contour that have the maximum/minimum values in x and y directions. The leftmost and rightmost points give the fingertips. The topmost point gives the top of the head. Since there are two feet, the contour needs to be divided by the vertical central axis and then the lowest point on each side of the contour gives the tip of the left foot or the tip of the right foot.

The neck point is needed in calculating the shoulder point (which is a turning point). The neck point is calculated as the point on the contour at the height of the neck location, where the neck location is calculated using local minimum from the previous steps.

The turning points, including the armpits, the crotch point, and the shoulder points, are located as the cuts that give the largest consistency after dividing a part of the contour between two adjacent border points. Due to the certain pose kept by the subject, the two adjacent body parts between the turning points always have a significant angle. (See figure 11.) The turning point should cut the silhouette in a way that the cutting result line strings should have maximum consistency internally. The cut that gives more consistency in the direction for each line string is more probable to be the turning point. For example, the cut at the crotch point divides the contour between the two feet into two parts. On the left part, the direction of two adjacent points on the contour should all go up; on the right part, the direction of two adjacent points on the contour should all go down. More in general, assuming a range of contour between point A and B can be divided into two parts in two different overall directions v_1 and v_2, the largest consistency cut k in a part of the contour as a series of points from p_1 to p_n can be located by finding the parameter k to maximize of the following equation:

$$\begin{aligned} &\arg\max \sum_{i=1}^{k} \overrightarrow{a_i} \cdot \overrightarrow{v_k} + \sum_{j=k+1}^{n} \overrightarrow{a_j} \cdot \overrightarrow{v_k}' \\ &\overrightarrow{a_i} = p_{i+1} - p_i \\ &\overrightarrow{v_k} = p_k - p_1 \\ &\overrightarrow{v_k}' = p_n - p_k \end{aligned} \tag{34}$$

The vectors $\overrightarrow{v_k}$ and $\overrightarrow{v_k}'$ are the direction vectors between the cut point and the end points of the line string. The vector $\overrightarrow{a_i}$ is the direction vector between two adjacent points on the contour.

The location of the heel at the side image is also located using the largest consistency cut. To find such a cut, two key locations need to be located first. On the side silhouette, we can easily find the border point on the foot as the points on the contour with largest (lowest) *y* value. Since OpenCV always constructs a contour following the counter-clockwise manner and the subject is facing right when taking the side image, to include the heel location, it is convenient to traverse from the index with the lowest point on the contour to points with smaller index in the contour. The traversal stops at the point on the contour having more than one-tenth of the height above the lowest point on the contour. This heuristic effectively finds a range on the contour roughly between the back of the ankle and the bottom of the foot. Then the largest consistency cut is applied on this range to locate the heel.

The height of the person is estimated from the side-view silhouette, as the vertical distance from the top of the head to the heel after the heel is located. The two armpits are used to estimate the width of the chest from the front silhouette. The shoulder points (combined with the neck points) are used to estimate the width of the shoulder.

The jacket location by definition is located as half way between the crotch point and the location of the waist.

4.4.3 Body Parts Location by Skeleton Analysis

The term "skeleton" is also referred as medial axis or topological skeleton of a binary image [Topological Skeleton, Wikipedia 2010]. The skeleton is calculated as the pixels in a binary image that have an equal distance to all the borders. Thus, the skeleton of an image represents the original binary image in the form of lines passing through the center of the silhouette, giving a look of a "skeleton." In this research, the algorithm is constructed using the online lecture notes from Wang [2004] by iterating through the following process until reaching convergence: for each non-zero pixel, the pixel location is assigned with the minimum value in a 3x3 neighborhood plus one. The skeleton points are the pixels with the centers being local maximum in the result of the above calculation.

The skeleton of the binary silhouette is used in locating the wrist on the arm by searching for the narrowest part on the lower arms. The lower arm is roughly located using the lower half of the bounding rectangle defined by the shoulder point and the fingertip point. The skeleton of the arm is used to calculate a direction of the arm using Hough transform. The direction perpendicular to the arm is used to trace from a skeleton point to two intersection points on the contour. The width of the arm is calculated from the two intersection points, and the skeleton point with the minimum in the second-order derivative of the width of the arm is considered the wrist (as to detect the wrist as the location of being narrow at the center and getting broad on its two sides).

This wrist location along with the shoulder point and armpits will be used to calculate the sleeve length and the sleeve inseam.

4.4.4 Ellipse Model in Measurement

The circumference *l* is estimated as the circumference of an ellipse, with the width measurement from the front image as the ellipse's major axis, and the width measurement from the side image as the ellipse' minor axis [Ward L. R. 2010]:

$$l = 2\pi\sqrt{\frac{a^2 + b^2}{2}} \tag{35}$$

The chest girth, the waist girth, the jacket girth, and the hip girth are all estimated using this equation, after reading *a* value and *b* value from the horizontal projection on the front and the side images. For example, from the side image, the hip location has been estimated. The same location for the hip will be referenced in the front image. The width of the body on the front image will be used as *a* and the width of the body in the side image will be used as *b*. Similarly, the neck girth, the hip girth and the jacket girth are calculated. Chest girth, however, cannot be measured in the front image for the width in horizontal projection due to the problem of having shoulders included in the horizontal projection. Instead, the width of the chest *a* on the front image is calculated as the Euclidean distance between the two armpits.

4.4.5 Triangular Model in Measurement

The neck girth can be estimated using ellipse (or circle). However, the measurement obtained from the ellipse model tends to be smaller than the actual measurement during the experiments. This is because the manual measuring of the neck reaches down to the joint of the collarbones, such that the shape of the tape measure is close to a triangle than an ellipse. To model this measurement, the joint of the collarbones is estimated at the middle of the two shoulder points. The neck points from the front view are estimated as the points on the contour that are closest to the backbone around the neck region. The width of the neck from the side view is taken to calculate the circumference of half of a circle, as modeling the back of the neck. The neck girth is calculated as the sum of the two distances between neck point and the joint of the collarbone plus half of a circle that represents the back of the neck.

4.4.6 Line Models in Measurements

The length of the body part is measured by the Euclidean distance between two key points. The sleeve length is calculated as the distance between the wrist point and the shoulder point plus half of the shoulder width. The shoulder width is measured as the length between a shoulder point and the intersection of the two shoulder-lines. The direction of the shoulder line is calculated as the line formed from the neck point and the shoulder point. In case the two shoulder lines are parallel to each other, the shoulder width is calculated as the distance between to shoulder points.

The sleeve inseam is calculated as the distance between the wrist and the armpit. Since there are two arms, the sleeve length and the sleeve inseam have two measurements. An average on the two values will be taken and used as the final measurement for sleeve length/inseam.

The pant inseam is calculated as the distance that the crotch point is above the ground. The height is calculated as the *y* distance between the top of the head and the location of the heel from the side silhouette. As mentioned earlier, the location of the ground is located as the location of the heel from the side silhouette.

4.5 Pixel Length to Inches

User will provide the height in inches of himself or herself. The length in inches of any other measurement is calculated by proportion to the height.

4.6 Image Acquisition

Since this is a research related to human subjects, at first, an Institutional Review Board (IRB) application was filed and approved to protect the participating subjects (IRB protocol number: 10-129). Voluntary participants were gathered from Cal Poly students and

faculties as the test subjects. The privacy of the subject was guaranteed during the data collection process by operating within a lab with locked door, with a female and a male researcher available to conduct the experiment. The images of the subjects were locked in a password-protected computer in the lab, with their faces erased. Each subject was assigned a code name to detach the identity. The images of the subjects will not be published. Only the numerical data and the silhouettes derived from the images can be published.

Three methods were used in collecting measurements: the newly developed single camera body measurement extraction system (as our method, described in the methodology), the TC^2 3D body scanner, and manual tape measure. The following data were collected: a front image and a side image of a subject (along with image subtraction results) were collected for the newly developed single camera body measurement extraction system, the 3D models of the subjects extracted from TC^2 3D body scanner. The nine measurements were extracted using a tape measure and a ruler. (See below for the definition of the nine measurements.)

4.7 Measurement Definition in the Experiment

Once the data was collected, the images were processed by the new human body measurement system to extract measurements. The following sizes were measured on the subject with the three methods: height, neck, chest, waist, jacket (high hip), seat, sleeve length, sleeve inseam, pant in seam. The definitions of the sizes are:

- Height: length in inches measured from the top of the head to the floor.
- Neck: circumference in inches as measured angled from neck base in back of the neck, to the pit of the neck at the front of the neck.
- Chest: circumference in inches around body, at the broadest part of the chest.

- Waist: circumference in inches around body at the narrowest part between chest and hip.
- Jacket: circumference in inches around body, as half way between the waist and the crotch. (Also defined as abdomen or high hip.)
- Seat: circumference in inches around body at the widest part of the hips as viewed from the side and the front.
- Sleeve Length: length in inches measured from the back neck bone along the curve of the back to the shoulder point, and then in a straight line to the end of the wrist.
- Sleeve Inseam: length in inches measured from the armpit to the end of the wrist, in a straight line.
- Pant Inseam: length in inches measured from the straight-line distance between the crotch point and floor.

4.8 Evaluation

The automatically extracted measurements using the two systems were compared with the measurements extracted from the manual tape measure. Then the error ratio was derived using the following a statistical analysis:

- The average error for the measurement of each part by comparing the measurement from the newly developed single camera body measurement extraction system and the sizes gathered from the manual tape measure
- The average error for the measurement of each part by comparing the measurement from the TC^2 3D body scanner system and the sizes gathered from the manual tape measure

4.9 Software Implementation

The software application is implemented in C# language using Microsoft .Net framework (version 3.5). The GUI (Graphics User Interface) is created using the GUI editor provided in Microsoft Visual C# 2008 Express Edition. In general, the GUI provides a test bed to run the algorithm.

EmguCV, which is a C# wrapper of OpenCV, is used in the implementation of this application. While OpenCV is an open source project developed by Intel, EmguCV can either be used under a GPL license or for commercial use with a small fee. EmguCV is mostly used in this project to perform in image subtraction, image median smooth, video capture, edge detection, Hough transform and contour extraction. EmguCV version 2.0 was downloaded from http://www.emgu.com in 2009. OpenCV 2.0 was also installed in order for EmguCV to invoke the functionality upon. (Open CV 2.0 was licensed under Open Source BSD license.)

DevCorp Colorspace library was downloaded from CodeProject.com to help converting RGB color to CIE Lab color [Leparmentier 2007]. The code is licensed under The Code Project Open License (CPOL) 1.02. The main points subject to the terms of the license are: source code and executable Files can be used in commercial applications; source code and executable files can be redistributed; and source code can be modified to create derivative works. No claim of suitability, guarantee, or any warranty whatsoever is provided. The software is provided "as-is". The article(s) accompanying the work may not be distributed or republished without the Author's consent.

Log4Net is used in the place of logging debugging information and errors, and it is under Apache License.

The algorithms for image segmentation, blob identification, key point estimation, background image statistics (for background subtraction), and measuring heuristics, are hand coded in C#.

Chapter 5

Results

The data from 31 test subjects were collected (19 females and 12 males). The data was gathered by the Apparel Merchandizing Management Department of California State Polytechnique University, Pomona. Three measuring methods (mentioned in the previous chapter) are used to measure the sizes of each subject. (For detailed measurements, refer to the appendix.) The results from the two automated measurements are compared to the manual measuring method as a benchmark. The average error rates are shown as below in table 1 and figure 12 for two automated measurement methods. The error rates considering each subject on each part are shown in figure 13 to 20.

Our new method is named as "WMM" (Web cam Measuring Method) in the following sections.

5.1 Error Rates Of Web Cam Measuring Method (WMM)

Except the sleeve inseam, the average error rates of our method (see table 1, column "WMM") are falling within 10%. Comparing to the existing automated body-measuring methods TC^2 (see figure 12), WMM has a leading accuracy at the chest circumference, while being close to TC^2 at waist circumference, seat circumference and sleeve length estimation. The new method (WMM) has an average error rate below 10% for seven parameters (except sleeve inseam), while three of them (chest circumference, seat circumference, and sleeve length) are below 5.1% of error.

	TC^2	WMM
Height		
Neck circumference	**3.96**	8.63
Chest circumference	5.83	**4.51**
Waist circumference	**5.65**	6.02
Jacket circumference	**4.16**	7.41
Seat circumference	**3.45**	3.98
Sleeve length	**4.58**	5.10
Sleeve inseam	**5.03**	15.51
Pant inseam	**7.47**	9.73

Table 1: Average error rates comparing to manual tape measurements. For a specific kind of measurement, the more accurate method is shown in bold.

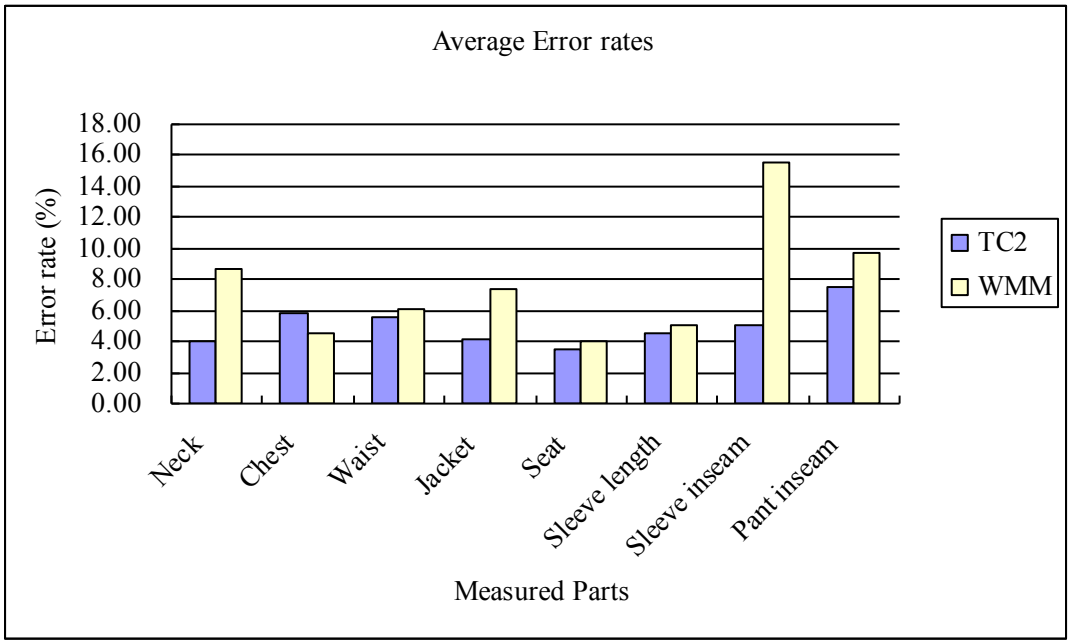

Figure 13: Comparing the average error rates of TC^2 against WMM.

5.2 Error Rates of TC^2 3D Body Scanner

The TC^2 3D body scanner achieves all average measurements below 10% of error (see table 1, column "TC^2"). It is most accurate at the neck circumference and seat circumference measurement (below 4% of error). Including the jacket circumference and the sleeve length, four types of measurements are below 5% of error. The pant inseam is the only measurement having an error rate above 7% while still being very reasonable under 10% of error.

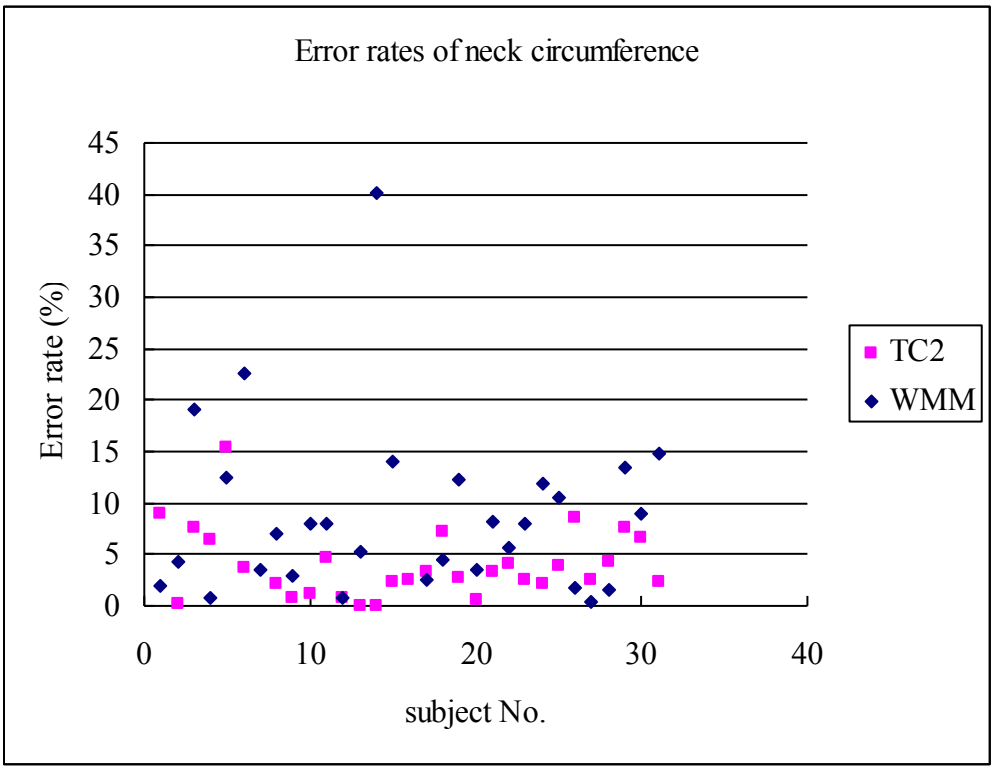

Figure 14: Error rates on measuring the neck circumference

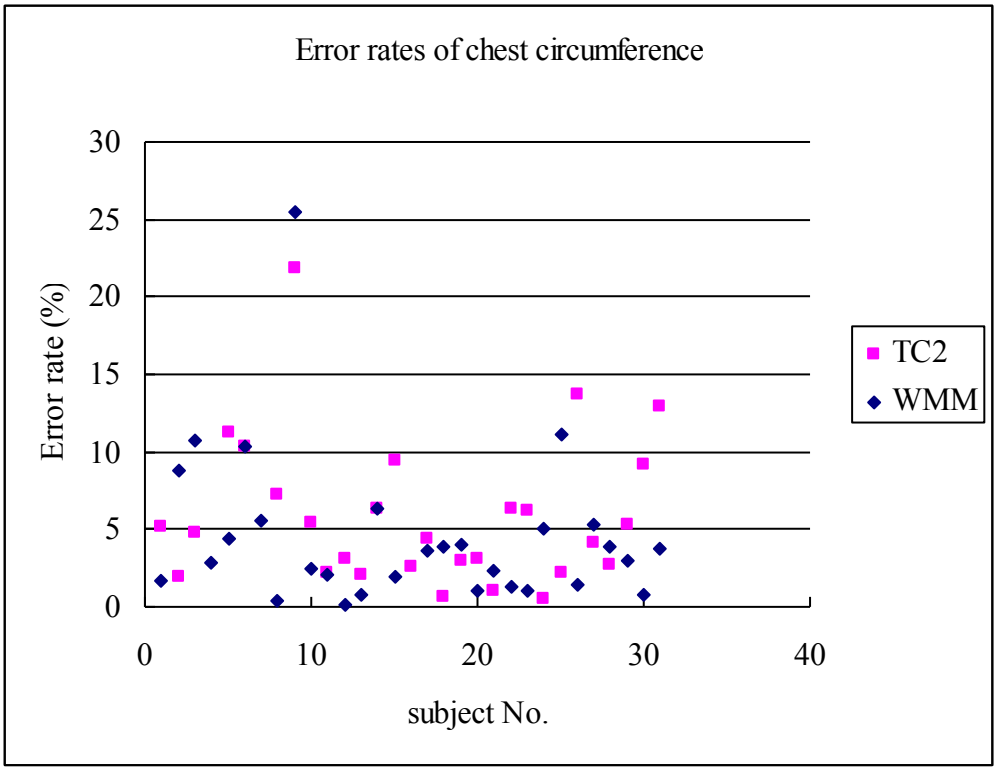

Figure 15: Error rates on measuring the chest circumference

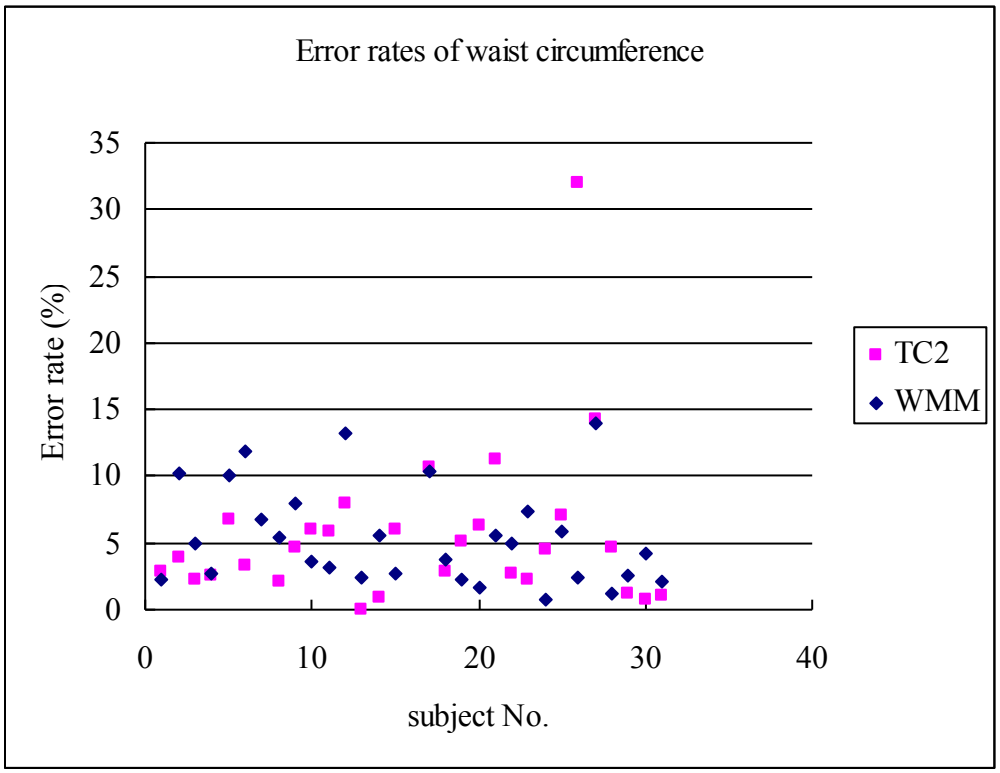

Figure 16: Error rates on measuring the waist circumference

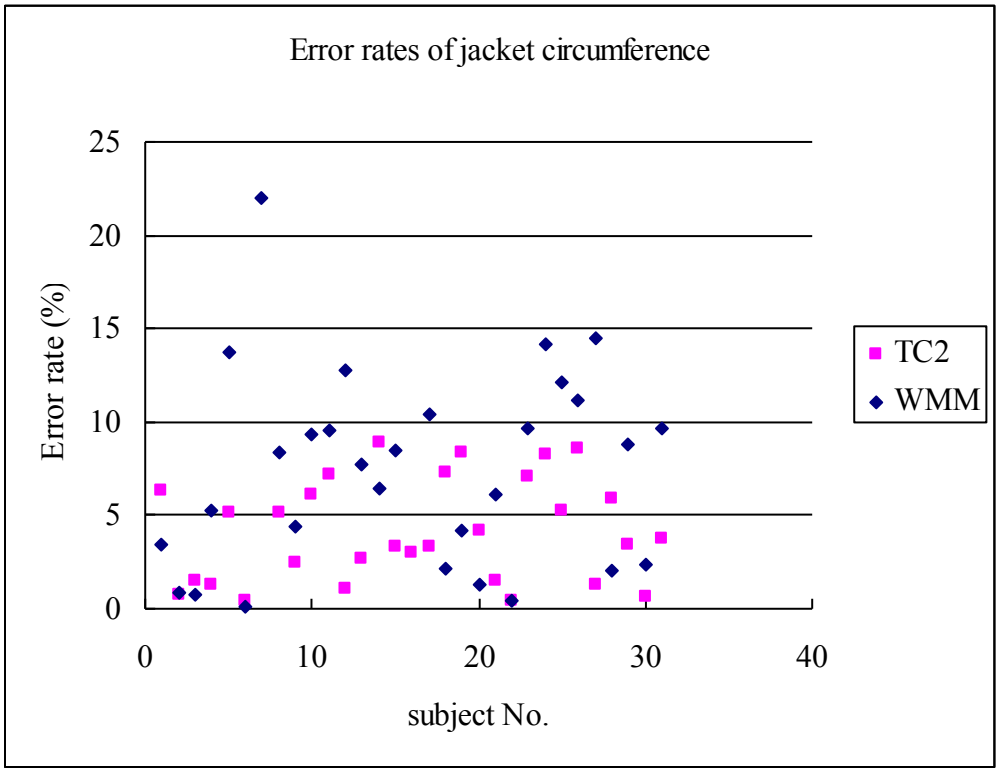

Figure 17: Error rates on measuring the jacket circumference

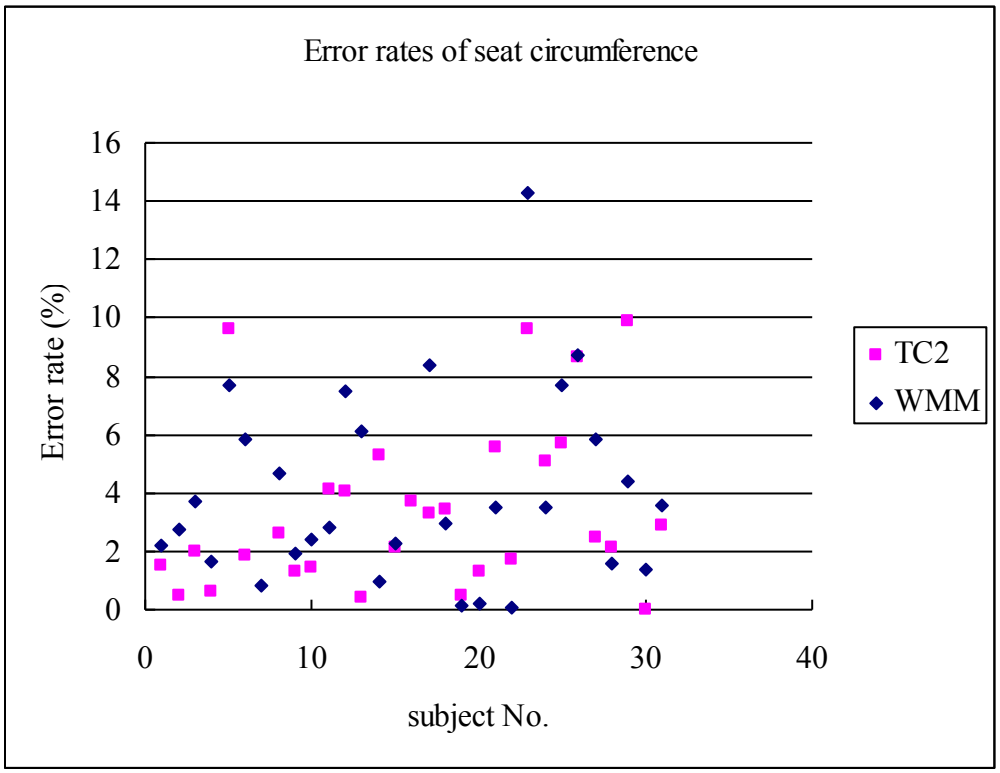

Figure 18: Error rates on measuring the seat circumference

Figure 19: Error rates on measuring the sleeve length

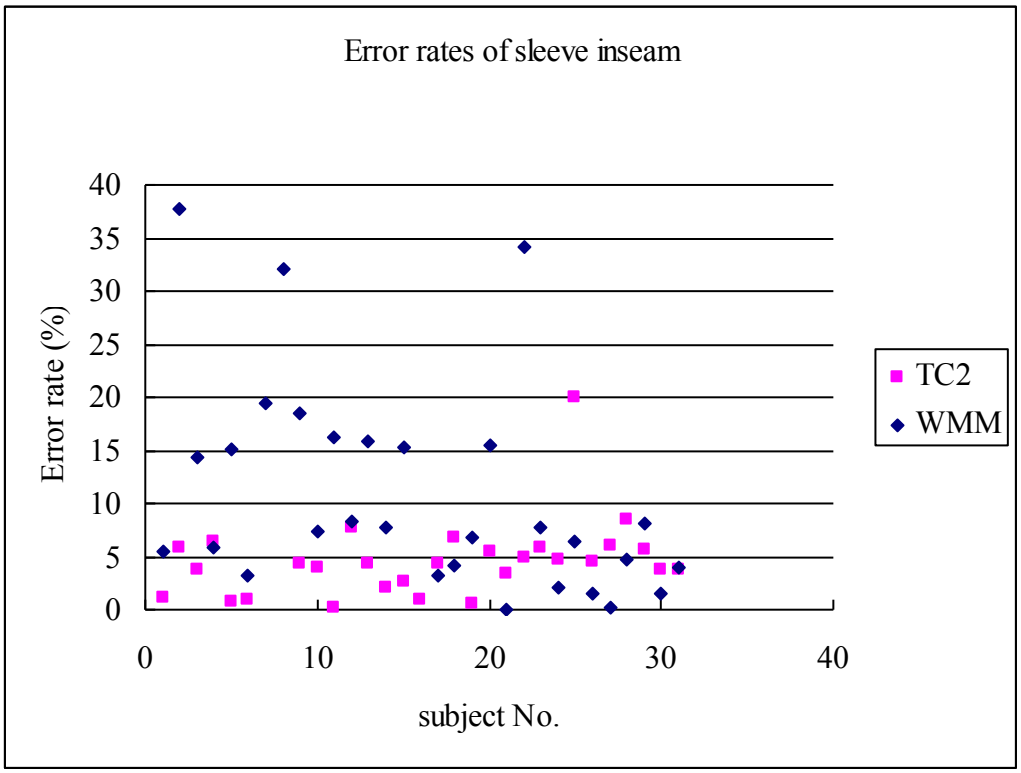

Figure 20: Error rates on measuring the sleeve inseam

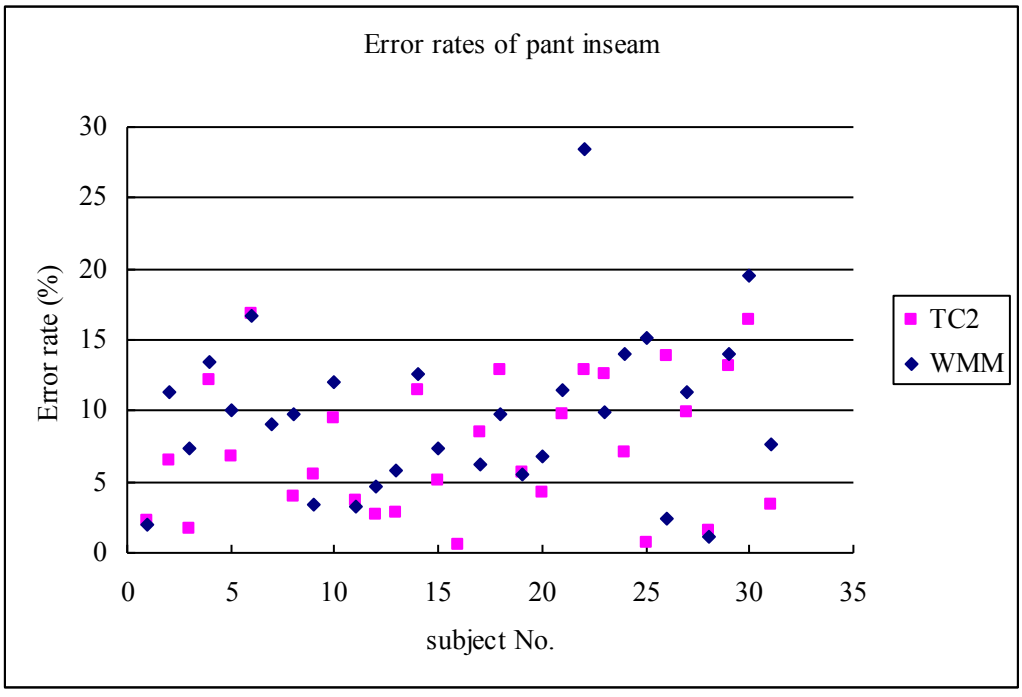

Figure 21: Error rates on measuring the pant inseam

Chapter 6

Discussion

6.1 Accuracy and source of errors

As it has shown in the results, though not as accurate as TC^2 3D body scanner, the performance of the single camera body measuring system can achieve a performance close to TC^2 3D body scanner, within 10% for most of the measurements.

As for our method WMM, the errors from neck circumference were usually caused by the error in segmentation. For some female subject, her ponytail overlapped with her neck, hiding the narrowest part of the neck from the measuring. Because the web cam was set to be lower than the subject's crotch point, for some tall subjects, their shoulders entered the neck region in the silhouettes, introducing errors to the results. Sleeve inseams were inaccurate because many subjects held their arms too closely to their bodies, closing the armpits inside the silhouette. Many male subjects wore loose boxers, making the crotch points lower to the actual position.

TC^2 3D body scanner in theory is very accurate. It was arguable that whether the benchmark should be the TC^2 3D body scanner or the manual measurements, since manual measurements were commonly approximated and depended on human factors. In addition, in the experiments, the manual measurements were obtained with the subjects in their daily clothes, while they were in tightly fitted clothes while being measured by TC^2 3D scanner and WMM. However, since the custom-fitted clothes were traditionally all made with the manual measurements and the manual method was the only way to measure physically, the manual tape measure was considered as the benchmark in this experiment.

6.2 Convenience to Use

In the two automated body-measuring methods, TC^2 3D body scanner requires the less user input: only a single mouse click to initiate the scanning process. The WMM system requires the operator to pick the good silhouettes among the recorded to start with and adjust parameters.

Since the background subtraction is constantly showing on the screen, the subjects are given the feedback to actively participant during the process. (Versus in the TC^2 system, the subject passively follows instructions from the operator.) The frame recording functionality gives the subjects the ability to be the operators themselves at the same time.

TC^2 is completely automated, taking about two minutes for the whole scanning process. The WMM method is automatically calibrated (in five seconds) and is fast for recording the silhouettes with little communication between the subject and the operator, requiring the least participation time from the subject.

A TC^2 body scanner costs tens of thousands of dollars for making the multiple calibrated cameras with the cubicle, while WMM only require a home PC with a cheap web cam that cost a few hundreds dollars in total. Moreover, WMM is portable. It can be easily setup or put aside. It can be carried along with a laptop, while TC^2 3D body scanner is huge and impossible to be transported without being disassembled.

Chapter 7

Conclusion and Future Development

In conclusion, with a simple web cam and a PC in a less restricted room environment, our method WMM achieved reasonable measurement performance for seven out of the eight parameters, while four of eight parameters can have a performance close to or better than the TC^2 3D body scanner. It proves that human body measurement extraction can be done using inexpensive equipment to have reasonable results.

However, since half of the parameters (neck circumference, jacket circumference, sleeve inseam and pant inseam) still have error rates over 7%, the single camera body measuring systems (WMM) still need further development. Ideally, the error rates for all the measurements should be within one inch to two inches (less than 4% of error for a narrow waist) in order to really make fitted clothes.

In future development, more sophisticated segmentation methods such as Bayesian matting and Graph Cut will be explored and tested to improve the quality of the silhouette. Various post processing can be tested to receive a more accurate while smoothed silhouette. In measuring method, more sophisticated methods need to be used in locating the key points, such that the neck circumference and the sleeve inseam measurements can tolerate errors introduced in the segmentation process. Body measurements other than the basic eight parameters can be extracted after more sophisticated segmentation methods are introduced to locate different parts of body.

Since the single camera body-measuring system is targeting to become a product that users can use from home, more works need to be done to make the system to work in a confined space with poor light sources. Shadow removal needs to be explored to loosen the constraints on the required lighting environment. More

performance testing needs to be conducted with subjects in front of more natural backgrounds.

Bibliography

Aldrich, W. *Metric Pattern Cutting for Menswear: Including Unisex Clothes and Computer Aided Design*, 3rd ed., Blackwell Science, 1997.

Aldrich, W. *Metric Pattern Cutting for Women's Wear*, 5th ed., Blackwell Publishing, 2008.

Apostoloff, N. and Fitzgibbon, A. "Bayesian Video Matting Using Learnt Image Priors." *IEEE Computer Society Conferences on Computer Vision and Pattern Recognition* (2004), vol. 1, 407-414.

Boykov, Y. and Funka-Lea, G. "Graph Cuts and Efficient N-D Image Segmentation." *International Journal of Computer Vision* (2006), vol. 70, no. 2, 109-131.

Boykov, Y. and Jolly, M. "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images." *Proceedings of Internation Conference on Computer Vision*, (2001), vol.1, 105-102.

Capo, A. J., Varona, J., Hidalgo, M.G. and Perales, F. J. "Automatic Human Body Modeling for Vision-based Motion Capture." *SHORT Papers proceedings: The 14th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision* (2006), 81-86.

Certain, A. and Stuetzle, W. "Automatic Body Measurement for Mass Customization of Garments." *Proceedings of the second International Conference on 3-D Imaging and Modeling, National Research Council of Canada, IEEE* (1999), 405-412.

Chang, F., Chen, C. and Lu, C. "A Linear-time Component-labeling Algorithm Using Contour Tracing Technique." *Computer Vision Image Understanding* (2004), vol. 93, no. 2, 206–220.

Chuang, Y., Curless, B., Salesin, D. and Szeliski, R. "A Bayesian Approach to Digital Matting." *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2001), vol. 2, 264-271.

Crabb, R., Tracey, C., Puranik, A., and Davis, J. "Real-time Foreground Segmentation via Range and Color Imaging." *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops on Time of Flight Camera Based Computer Vision* (2008), 1-5.

Crow, F. "Summed-area Tables for Texture Mapping." *SIGGRAPH '84: Proceedings of the 11th Annual Conference on Computer graphics and interactive techniques* (1984), vol. 18, no. 3, 207–212.

Elgammal, A., Harwood, D. and Davis L. 2000. "Non-parametric Model for Background Subtraction." *FRAME-RATE Workshop, IEEE* (2000), vol. 1843, 751-767.

Gordon, D. H., Darrell, T., Harville, M., and Woodfill J. "Background Estimation and Removal Based on Range and Color." *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (June, 1999), vol. 2, 459-464.

Green, B. 2002. "Canny Edge Detection Tutorial." *Autonomous Systems Lab.* Drexel University. http://www.pages.drexel.edu/~weg22/can_tut.html (accessed, August 4, 2010).

Hoffman, D. D. and Singh, M. "Salience of Visual Parts." *Cognition* (1997)*,* vol. 63, no. 1, 29-78.

Jung, C. R. "Efficient Background Subtraction and Shadow Removal for Monochromatic Video Sequences." *IEEE Transactions on Multimedia* (April 2009), vol. 11, no. 3, 571-577.

Kologorov, V. and Zabih, R. "What Energy Functions Can be Minimized via Graph Cuts?" *IEEE Transactions on Pattern Analysis and machine Intelligence* (Feb 2004), vol. 26, no. 2, 147-159.

Latecki, L.J. and Lakaemper, R. "Convexity Rule for Shape Decomposition Based on Discrete Contour Evolution." *Computer Vision and Image Understanding* (1999), vol. 73, no. 3, 441-454.

Lee, Y. T. *Body Dimensions for Apparel*, NISTIR 5411, National Institute of Standards and Technology, 1994.

Leparmentier, G. "Manipulating Colors in .Net - Part 1". June 3, 2007. The Code Project. Available from http://www.codeproject.com/KB/recipes/colorspace1.aspx. (accessed August 21, 2010). Source Code.

Matusik, W. *Image-based Visual Hulls: Master of Science in Computer Science and Engineering at the MASSACHUSETTS INSTITUTE OF TECHNOLOGY*. Massachusetts Institute of Technology, 2001.

Meunier, P. and Yin, S. "Performance of a 2D Image-based Anthropometric Measurement And Clothing Sizing System". *Applied Ergonomics* (2000), vol. 31, no. 5, 445-451.

Milton, J. S. & Arnold, J. C. *Introduction To Probability and Statistics*. 4th ed. McGraw-Hill, New York, 2003.

Mittal, A., Zhao, L., and Davis, L. S. "Human Body Pose Estimation Using Silhouette Shape Analysis." *Proceedings of the Conference on Advanced Video and Signal Based Surveillance (AVSS'03),* Miami, FL (2003), 263–270.

Morse, B. S., Liu, W., Yoo, T. S. and Subramanian, K. "Active Contours Using a Constraint-Based Implicit Representation." *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2005), vol. 1, 285-292.

Paris, S. and Durand, F. "A Fast Approximation of the Bilateral Filter using a Signal Processing Approach." *Proceeding of European Conference on Computer* Vision (2006), vol. 81, no. 24-52.

Pfister, H. Camera Measurement System. *US Patent No.: 6,490,534* (Dec 3, 2002).

Peng, T. and Gupta, S.K. "Model and Algorithms for Point Cloud Construction using Digital Projection Patterns." *Journal of Computing and Information Science in Engineering* (2007), vol. 7, no. 4, 372-381.

Reinhard, E. & Khan, E. "A. Depth-of-field-based Alpha-matte Extraction." *Proceedings of the 2nd Symposium on Applied Perception in Graphics and Visualization* (2005), 95-102.

Ren, M., Yang, J., and Sun, H. "Tracing Boundary Contours in a Binary Image." *Image Vision and Computing* (2002), vol. 20, 125-131.

Rittscher, J., Kato, J., Joga, S. and Blake, A. "A Probabilistic Background Model for Tracking." *Proceedings of the 6th European Conference on Computer Vision-Part II. Lecture Notes In Computer Science*: vol. 1843, (2000), vol. 2, 336-350.

Rother, C., Kolmogorov, V. and Blake, A. "GrabCut – Interactive Foreground Extraction using Iterated Graph Cuts". *ACM SIGGRAPH* (2004), vol. 23, no. 3, 309-314.

Singh, M., Seyranian, G. D., and Hoofman, D. D. "Parsing Silhouettes: The Short-cut Rule." *Perception and Psychophysics* (1999), vol. 61, no. 4, 636-660.

Sun, J., Jia, J., Tang, C., and Shum, H. "Poisson Matting." *International Conference on Computer Graphics and Interactive Techniques*, *ACM SIGGRAPH* (2004), vol. 23, 315-321.

Sun, J., Li, Y., Kang, S., and Shum, H. "Flash Matting." *Proceedings of ACM SIGGRAPH 2006*, *ACM Transactions on Graphics (TOG)* (July 2006), vol. 25, no. 3, 772-778.

Tomasi, C. and Manduchi, R. 1998. "Bilateral Filtering for Gray and Color Images." *Sixth International Conference on Computer Vision* (1998), 839-846.

Tuit, A. *Introducing Pattern Cutting*, Heinemann Educational Books, London, 1974.

Viola, P. and Jones, M. "Robust Real-time Object Detection." *International Journal of Computer Vision* (2001).

Wang, C.C.L., Wang, Y., Cheng, T.K.K., and Yuen, M.M.F. "Virtual Human Modeling from Photographs for Garment Industry." *Computer-Aided Design* (2003), vol. 35 no. 6, 577–589.

Wang, C.C.L., Wang, Y., and Yuen M.M.F. "Feature Based 3D Garment Design Through 2D Sketches." *Computer-Aided Design* (2003), vol. 35, no.7, 659-672.

Wang, R. "Mathematical Morphology: A Skeletonization Algorithm." (November 2, 2004). http://fourier.eng.hmc.edu/e161/lectures/morphology/node3.html. (accessed July 1, 2010).

Ward, R. L. "Circumference of an Ellipse." The Math Forum @ Drexel, 2010. http://mathforum.org/dr/math/faq/formulas/faq.ellipse.circumference.html. (accessed August 19, 2010).

Wikipedia. "Chroma key." Wikipedia.org. (revised August 20, 2010), http://en.wikipedia.org/wiki/Chroma_key. (accessed August 20, 2010).

Wikipedia. "*CIE 1931 Color Space.*" Wikipedia.org. (revised August 13, 2010), http://en.wikipedia.org/wiki/CIE_XYZ_color_space (accessed Feb 4, 2010).

Wikipedia. "Divergence." Wikipedia.org. (revised August 9, 2010), http://en.wikipedia.org/wiki/Divergence. (accessed August 9, 2010).

Wikipedia. "Gaussian Function." Wikipedia.org. (revised August 4, 2010), http://en.wikipedia.org/wiki/Gaussian_function (accessed August 5, 2010).

Wikipedia. "Lab Color Space. " Wikipedia.org. (revised August 4, 2010), http://en.wikipedia.org/wiki/Gaussian_function. (accessed Feb 4, 2010).

Wikipedia. "Structured-light 3D scanner. " Wikipedia.org. (revised August 16, 2010), http://en.wikipedia.org/wiki/Structured-light_3D_scanner. (accessed August 3, 2010).

Wikipedia. "Topological Skeleton. " Wikipedia.org. (revised May 25, 2010), http://en.wikipedia.org/wiki/Topological_skeleton. (accessed July 1, 2010).

Yamauchi, K. and Sata, Y. "3D Human Body Measurement by Multiple Range Images." *The 18th International Conference on Pattern Recognition* (2006), vol. 4, 833-836.

Zheng, Y., Kambhamettu, C., Yu, J., Bauer, T, and Steiner, K. "FuzzyMatte: A Computationally Efficient Scheme for Interactive Matting." *IEEE Conference On Computer Vision and Pattern Recognition* (2008), 1-8.

Zhong, Y. and Xu, B. "Automatic Segmenting and Measurement on Scanned Human Body." *International Journal of Clothing Science and Technology* (2006), vol. 18, no.1, 19-30.

Appendix

H: Height
N: Neck
C: chest
W: waist
J: jacket
S: seat
SL: sleeve length
SI: sleeve inseam
P: pant inseam

Table 2: Measurements from the new single camera body measurement system "WMM" (our method)

	N	C	W	J	S	SL	SI	P
01	15.7	37.64	31.44	38.27	40.11	28.57	20.05	29.56
02	16.68	31.91	27.18	35.19	35.49	29.9	20.55	28.17
03	14.57	36.8	34.72	39.3	39.49	28.26	15.22	25.47
04	15.13	34.73	29.52	41.26	41.37	33.53	21.17	28.22
05	14.02	33.41	30.22	36.96	37.16	30.88	21	27.42
06	13.15	34.95	32.3	41.44	41.44	28.55	16.5	27.5
07	13.5	35.89	31.45	26.53	36.55	32.2	22.7	30.69
08	14.97	31.61	28.18	34.67	36.11	30.55	23.46	27.75
09	13.58	32.41	29.16	33.94	35.32	29.02	13.44	27.54
10	12.87	30.25	26	36.34	37.38	29.78	19.07	27.29
11	13.33	31.6	26.58	33.14	33.94	27.91	18.89	28.04
12	13.11	31.96	28.39	35.23	36	28.31	18.15	26.22
13	14.2	35.49	29.76	37.7	38.46	30.22	20.27	28.24
14	10.19	36.55	31.41	39.38	39.38	31.63	21.81	27.99
15	13.77	34.31	30.62	36.07	37.58	33.27	23.06	31.28
16								
17	13.15	31.33	27.57	35.62	36.3	30.02	18.83	29.32
18	13.85	33.65	29.57	36.78	38.08	27.28	17.01	26.16
19	11.85	31.19	27.54	36.19	38.06	28.05	17.62	28.83
20	13.51	34.16	27.96	35.52	37.08	26.73	18.19	26.35
21	12.86	31.76	27.65	36.36	36.48	28.46	18.5	28.55
22	15.57	38.5	35.86	41.82	43.04	26.87	10.53	20.05
23	13.81	35.62	33.28	41.11	42.86	28.25	15.69	27.03
24	12.78	32.29	29.17	38.25	39.34	27.28	16.6	26.21
25	12.07	38.33	29.65	38.39	39.04	27.18	17.09	23.76
26	13.73	30.56	24.6	34.47	34.79	29.49	18.97	29.27
27	12.94	33.68	29.58	35.5	35.98	26.98	16.96	25.28
28	13.29	31.7	28.04	33.66	35.19	26.29	16.45	26.71
29	14.29	40.15	35.08	41.36	40.97	31.03	22.72	27.94
30	15.01	37.71	32.34	38.07	40.03	31.04	20.57	27.36
31	16.65	31.12	28.09	32.9	32.79	31.99	18.74	28.16

Table 3: Measurements from TC^2 3D body scanner:

Subject	Neck	Chest	Waist	Jacket	Seat	Sleeve Length	Sleeve Inseam	Pant Inseam
01	14.58	38.9	31.37	39.33	41.56	29.67	18.515	29.66
02	16.04	35.69	28.36	35.78	37.47	31.1	19.915	29.69
03	16.64	43.22	36.8	39.59	40.18	30.545	17.77	27.98
04	15.95	37.62	30.42	39.7	41.22	31.37	19.505	28.65
05	13.55	35.6	29.1	34.18	38.89	30.26	19.36	32.58
06	16.36	43.04	34.33	41.7	45.67	30.735	17.215	27.44
07								
08	13.69	34.05	27.04	33.66	36.25	29.9	19.46	29.53
09	13.88	34.02	29.06	34.62	37.33	26.325	15.87	26.94
10	13.84	32.68	27.03	35.28	39.16	27.375	17.49	28.07
11	13.81	32.94	27.54	32.42	35.79	28.04	18.19	27.92
12	12.9	31.02	26.19	31.59	35.9	25.84	16.86	26.78
13	15.01	36.49	30.51	35.93	39.2	29.15	18.47	29.14
14	17.01	41.48	32.68	40.28	44.26	33.275	20.48	28.34
15	16.36	38.29	31.79	34.36	39.79	31.64	19.355	32.03
16	16.07	40.53	34.45	39.9	40.98	29.72	17.285	28.84
17	13.06	31.06	27.13	33.31	36.97	27.98	18.89	28.61
18	13.45	35.21	28.8	33.37	35.93	27.99	18.17	32.75
19	13.12	33.48	27.31	34.61	40.18	29.1	18.37	28.78
20	13.93	35.57	29.22	34.49	37.91	27.18	16.68	27.07
21	13.53	32.82	27.83	34.76	38.65	28.73	18.39	29.11
22	15.81	40.41	34.41	41.81	44.8	26.61	15.41	24.4
23	14.62	38.24	32.21	40.17	41.68	28.03	17.33	26.23
24	14.82	34.18	29.26	36.27	41.64	27.375	17.38	28.35
25	14.02	35.28	30.52	36.05	39.61	27.96	18.68	27.79
26	14.67	35.24	32.34	33.66	38.37	29.85	18.25	25.82
27	13.32	30.69	27.44	31.4	35.26	26.805	17.78	31.33
28	12.91	33.88	28.78	34.96	37.8	25.865	16.32	26.59
29	17.76	41.09	35.92	39.32	43.13	33.265	20.56	28.23
30	17.59	41.48	32.75	39.24	41.37	33.185	20.075	28.42
31	14.85	33.89	27.29	31.12	33.97	30.065	19.28	29.46

Table 4: Measurement from manual tape measure

	H	N	C	W	J	S	SL	SI	P
01	64.75	14	37	30.5	37	41	30	19	29
02	69.75	16	35	29.5	35.5	36.5	33	19	31.75
03	67	18	41.25	36	39	41	31.75	17.75	27.5
04	71	15	35.75	31.2	39.2	40.7	33.5	20	32.6
05	68	16	32	27.25	32.5	34.5	30.5	18.25	30.5
06	71	17	39	35.5	41.5	44	31	16	33
07	67	14	34	28.5	34	36.25	31.75	19	33.75
08	64	14	31.75	26.5	32	34.5	30.25	17.75	30.75
09	62	14	34.5	27.75	35.5	36	27.5	16.5	28.5
10	65.5	14	31	25.5	33.25	36.5	28.5	17.75	31
11	63	14.5	32.25	26	30.25	33	28	16.25	29
12	60	13	32	24.25	31.25	33.5	28	16.75	27.5
13	64.5	15	35.75	30.5	35	36.25	30.5	17.5	30
14	70.75	17	39	33	37	39	34	20.25	32
15	70.25	16	35	30	33.25	36.75	32.5	20	33.75
16	67.5	16.5	39.5	37.25	38.75	41.5	30	17	29
17	65.25	13.5	32.5	24.5	32.25	33.5	29.25	18.25	31.25
18	62.5	14.5	35	28	36	37	30	17.75	29
19	65.5	13.5	32.5	26	37.75	38	29.25	16.5	30.5
20	63.25	14	34.5	27.5	36	37	28.75	15.75	28.25
21	66.5	14	32.5	25	34.25	35.25	29.75	18.5	32.25
22	61	16.5	38	33.5	42	43	28	16	28
23	63.75	15	36	31.5	37.5	37.5	29.75	17	30
24	65.25	14.5	34	28	33.5	38	28.75	16.25	30.5
25	61.25	13.5	34.5	28.5	34.25	36.25	30.5	18.25	28
26	64.75	13.5	31	24.5	31	32	31.25	19.25	30
27	61.75	13	32	24	31	34	28.5	17	28.5
28	58.25	13.5	33	27.5	33	35.75	28.25	17.25	27
29	71.25	16.5	39	35.5	38	39.25	35.25	21	32.5
30	71.5	16.5	38	32.5	39	39.5	34.5	20.25	34
31	66.5	14.5	30	27	30	34	31.25	19.5	30.5