3

3D Ear Detection from Side Face Range Images

Human ear detection is the first task of a human ear recognition system and its performance significantly affects the overall quality of the system. In this chapter, we propose three techniques for locating human ears in side face range images: template matching based detection, ear shape model based detection, and fusion of color and range images and global-to-local registration based detection. The first two approaches only use range images, and the third approach fuses the color and range images.

3.1 3D Ear Detection Using Only Range Images

The template matching based approach [35] has two stages: offline model template building and online ear detection. The ear can be thought of as a rigid object with much concave and convex areas. The averaged histogram of shape index represents the ear model template since shape index is a quantitative measure of the shape of a surface. During the online detection, we first perform the step edge computation and thresholding since there is a sharp step edge around the ear boundary, and then we do image dilation and connected-component analysis to find the potential regions containing an ear. Next, for every potential region, we grow the region and compute the dissimilarity between each region's histogram of shape indexes and the model template. Finally, among all of the regions, we choose the one with the minimum dissimilarity as the detected region that contains ear.

For the second approach, the ear shape model is represented by a set of discrete 3D vertices corresponding to ear helix and anti-helix parts

[36]. Since the two curves formed by the ear helix and anti-helix parts are similar for different people, we do not take into account the small deformation of two curves between different persons, which greatly simplifies our ear shape model. Given side face range images, first the step edges are extracted; then the edge segments are dilated, thinned and grouped into different clusters which are the potential regions containing an ear. For each cluster, we register the ear shape model with the edges. The region with the minimum mean registration error is declared as the detected ear region; the ear helix and anti-helix parts are identified in this process.

3.1.1 Template Matching Based Detection

Shape Index

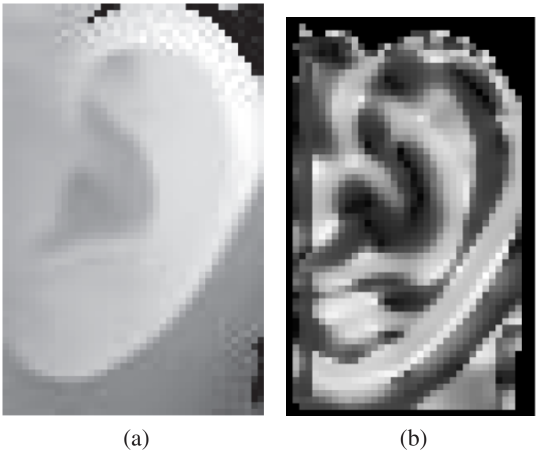
Shape index S_i, a quantitative measure of the shape of a surface at a point p, is defined by (3.1),

$$S_i(p) = \frac{1}{2} - \frac{1}{\pi}\tan^{-1}\frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)} \tag{3.1}$$

where k_1 and k_2 are maximum and minimum principal curvatures, respectively [60]. With this definition, all shapes can be mapped into the interval $S_i \in [0, 1]$. The shape categories and corresponding shape index ranges are listed in Table 3.1. From the table, we can see that larger shape index values represent convex surfaces and smaller shape index values represent concave surfaces.

Table 3.1. Surface shape categories and the range of shape index values.

Shape category	S_i **range**
Spherical cup	[0, 1/16)
Trough	[1/16, 3/16)
Rut	[3/16, 5/16)
Saddle rut	[5/16, 7/16)
Saddle	[7/16, 9/16)
Saddle ridge	[9/16, 11/16)
Ridge	[11/16, 13/16)
Dome	[13/16, 15/16)
Spherical cap	[15/16, 1]



(a) (b)

Fig. 3.1. (a) Ear range image. Darker pixels are away from the camera and the lighter ones are closer. (b) Its shape index image. The darker pixels correspond to concave surfaces and the lighter ones correspond to convex surfaces.

The ear has significant convex and concave areas, which gives us a hint to use the shape index for ear detection. The original ear range image and its shape index image are shown in Figure 3.1. In Figure 3.1(b), the brighter pixels denote large shape index values which correspond to ridge and dome surfaces. The ridge and valley areas form a pattern for ear detection. We use the distribution of shape index as a robust and compact descriptor since 2D shape index image is much too detailed. The histogram h can be calculated by $h(k) = \#\ of\ points\ with\ shape\ index\ \in bin(k)$. The histogram is normalized during the implementation.

Curvature Estimation

In order to estimate curvatures, we fit a biquadratic surface (3.2) to a local window and use the least squares method to estimate the parameters of the quadratic surface, and then use differential geometry to calculate the surface normal, Gaussian and mean curvatures and principal curvatures [49, 61]. Based on differential geometry, surface normal $\mathbf{n}$, Gaussian curvature K, mean curvature H, principal curvatures $k_{1,2}$ are given by (3.3), (3.4), (3.5) and (3.6), respectively:

$$f(x, y) = ax^2 + by^2 + cxy + dx + ey + f \qquad (3.2)$$

$$\mathbf{n} = \frac{(-f_x, -f_y, 1)}{\sqrt{1 + f_x^2 + f_y^2}} \tag{3.3}$$

$$K = \frac{f_{xx} f_{yy} - f_{xy}^2}{(1 + f_x^2 + f_y^2)^2} \tag{3.4}$$

$$H = \frac{f_{xx} + f_{yy} + f_{xx} f_y^2 + f_{yy} f_x^2 - 2 f_x f_y f_{xy}}{2(1 + f_x^2 + f_y^2)^{1.5}} \tag{3.5}$$

$$k_{1,2} = H \pm \sqrt{H^2 - K} \tag{3.6}$$

Model Template Building

Given a set of training side face range images, first we extract ears in each of the images manually and then calculate its shape index image and histogram the shape index image. After we get the histograms for each training image, we average the histograms and use the averaged histogram as our model template. Figure 3.2 shows the model template, obtained using 20 training images, in which the two peaks correspond to the convex and concave regions of the ear, respectively.

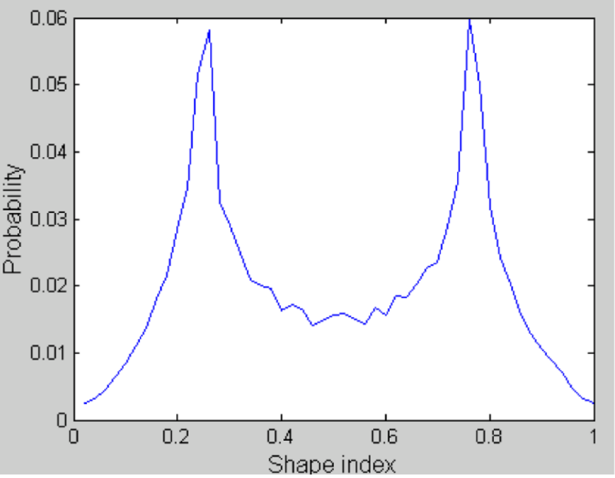


Fig. 3.2. Model template (discretized into 50 bins).

Step Edge Detection, Thresholding, and Dilation

There is a sharp change in depth around the ear helix part, which is helpful in identifying the ear region. Given a side face range image, the step edge magnitude, denoted by I_{step}, is calculated. I_{step} is defined by the maximum distance in depth between the center pixel and its neighbors in a $w \times w$ window. I_{step} can be written as:

$$I_{\text{step}}(i,j) = \max|z(i,j) - z(i+k, j+l)|, \\ -(w-1)/2 \leq k, l \leq (w-1)/2 \qquad (3.7)$$

where w is the width of the window and $z(i,j)$ is the z coordinate of the point (i,j). To get the step edge magnitude image, a $w \times w$ window is translated over the original side face range image and the maximum distance calculated from (3.7) replaces the pixel value of the pixel covered by the center of the window. The original side face range image and its step edge magnitude image are shown in Figure 3.3(a) and (b). From Figure 3.3(b), we clearly see that there is a sharp step edge around the ear boundary since brighter points denote large step edge magnitude.

The step edge image is thresholded to get a binary image which is shown in Figure 3.3(c). The threshold is set based on the maximum of I_{step}. Therefore, we can get a binary image by using (3.8),

$$F_T(i,j) = \begin{cases} 1 & \text{if } I_{\text{step}}(i,j) \geq \alpha * \max\{I_{\text{step}}\} \\ & \quad 0 \leq \alpha \leq 1 \\ 0 & \text{otherwise} \end{cases} \qquad (3.8)$$

There are some holes in the thresholded binary image and we want to get the potential regions containing ears. We dilate the binary image to fill the holes. The dilated image is shown in Figure 3.3(d). There are some holes in the thresholded binary image and we would like to get the potential regions containing ears. We dilate the binary image to fill the holes using a 3×3 structuring element. The dilated image is shown in Figure 3.3(d).

Connected Component Labeling

Using the above result, we proceed to determine which regions can possibly contain human ears. To do so, we need to determine the

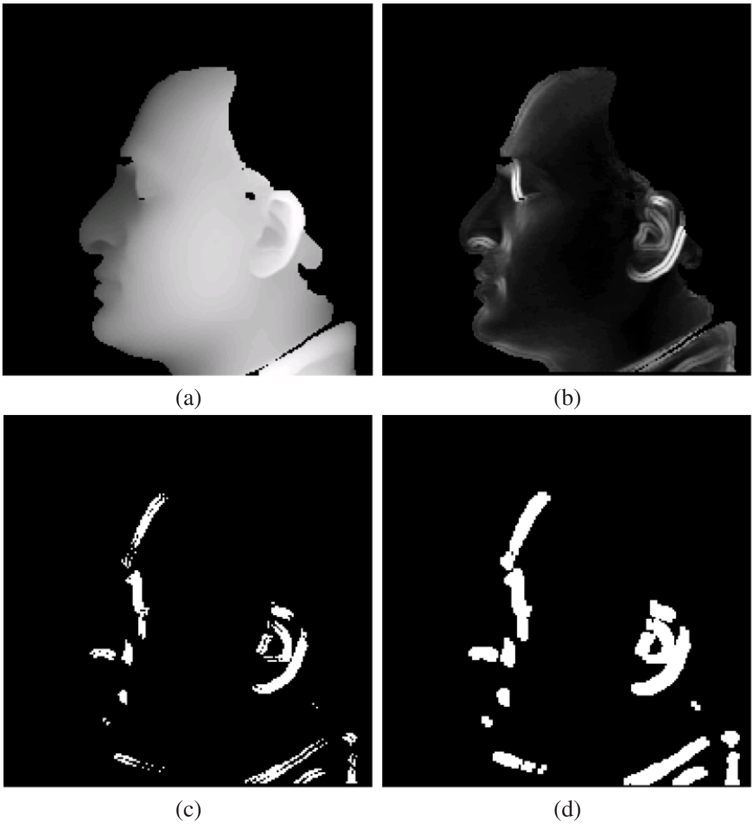


Fig. 3.3. (a) Original side face range image. (b) Step edge magnitude image. (c) Thresholded binary image. (d) Dilated image.

number of potential regions in the image. By running the connected component labeling algorithm, we can determine the number of regions. We used an 8-connected neighborhood to label a pixel. We remove smaller components whose area are less than β since the ear region is not small. The labeling result is shown in Figure 3.4(a) and the result after removing smaller components is shown in Figure 3.4(b).

After we get regions, we need to know their geometric properties such as the position and orientation. The position of a region may be defined using the center of the region. The center of area in binary images is the same as the center of the mass and it is computed as follows:

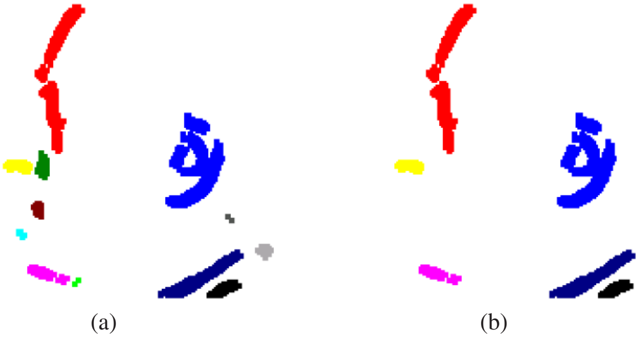


Fig. 3.4. (a) Labeled image. (b) Labeled image after removing smaller components.

$$\bar{x} = \frac{1}{A}\sum_{i=1}^{n}\sum_{j=1}^{m} jB[i,j], \quad \bar{y} = \frac{1}{A}\sum_{i=1}^{n}\sum_{j=1}^{m} iB[i,j] \tag{3.9}$$

where B is $n \times m$ matrix representation of the binary region and A is the size of the region. For the orientation, we find the axis of elongation of the region. Along this axis the moment of the inertia will be the minimum. The axis is computed by finding the line for which the sum of the squared distances between region points and the line is minimum. The angle of θ is given by (3.10):

$$\theta = \frac{1}{2}\tan^{-1}\frac{b}{a-c} \tag{3.10}$$

The parameters a, b and c are given by (3.11), (3.12), and (3.13), respectively.

$$a = \sum_{i=1}^{n}\sum_{j=1}^{m} (x'_{ij})^2 B[i,j] \tag{3.11}$$

$$b = 2\sum_{i=1}^{n}\sum_{j=1}^{m} x'_{ij}y'_{ij} B[i,j] \tag{3.12}$$

$$c = \sum_{i=1}^{n}\sum_{j=1}^{m} (y'_{ij})^2 B[i,j] \tag{3.13}$$

where $x' = x - \bar{x}$ and $y' = y - \bar{y}$. θ gives us a hint about the region growing direction.

Template Matching

As mentioned in Section 3.1.1, the model template is represented by an averaged histogram of shape index. Since a histogram can be thought of as an approximation of a probability distribution function, it is natural to use the $\chi^2 - divergence$ function (3.14) [62],

$$\chi^2(Q,V) = \sum_i \frac{(q_i - v_i)^2}{q_i + v_i} \tag{3.14}$$

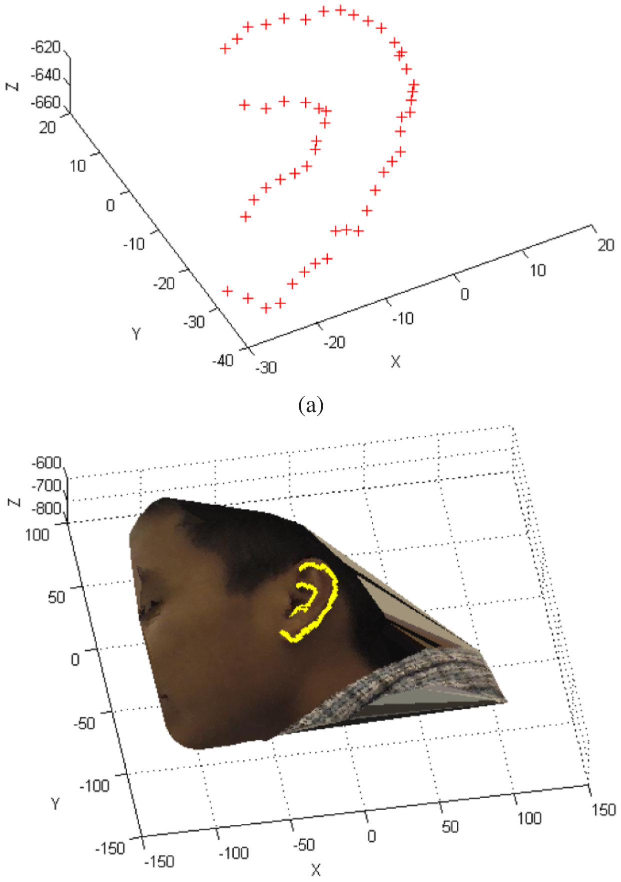
where Q and V are normalized histograms. From (3.14), we know the dissimilarity is between 0 and 2. If the two histograms are exactly the same, the dissimilarity will be zero. If the two histograms do not overlap with each other, it will achieve the maximum value 2.

From Section 3.1.1, we get the potential regions that may contain the ears. For each region, we can find a minimum rectangular bounding box to include the region; then we grow the region based on the angle θ. If $0 \leq \theta \leq \pi/2$, we grow the rectangle by moving the top-right vertex right, up, and anti-diagonal, and then moving the bottom-left vertex left, down, and anti-diagonal. If $\pi/2 \leq \theta \leq \pi$, we grow the rectangle by moving the top-left vertex left, up, and diagonal, and then moving the bottom-right vertex right, down, and diagonal. For each region, we choose the grown rectangular box with the minimum dissimilarity as the candidate ear region. Finally, over all of the candidate regions, we select the one with the minimum dissimilarity as the detected ear region. We set a threshold γ for region growing, which controls the size of the region.

3.1.2 Reference Shape Model Based Ear Detection

Ear Shape Model Building

Considering the fact that the curves formed by ear helix and anti-helix parts are similar for different people, we construct the ear shape model from one person only. The reference ear shape model s is defined by 3D coordinates $\{x, y, z\}$ of n vertices which lie on the ear helix and the anti-helix parts. The ear helix and the anti-helix parts are manually marked for the reference shape model. The shape model s is represented by a $3n \times 1$ vector $(x_1, y_1, z_1, x_2, y_2, z_2, \cdots, x_n, y_n, z_n)^T$. Figure 3.5(a) shows the ear shape model s marked by the pluses (+). The corresponding color image is also shown in Figure 3.5(b).



(a)

(b)

Fig. 3.5. The reference ear shape model. (a) The reference 3D ear shape model is displayed by the pluses (+). (b) Ear shape model is overlaid on the textured 3D face. The units of x, y and z are in mm.

Step Edge Detection and Thresholding

Given the step face range image, the step edge magnitude can be calculated as described in Section 3.1.1. One example of step edge magnitude image is shown in Figure 3.6. Figure 3.6(a) shows the original side face range image. In Figure 3.6(b), larger magnitudes are displayed as brighter pixels. We can clearly see that most of the step edge magnitudes are small values. To get edges, the step edge magnitude image must be segmented using a threshold operator. The selection of threshold value is based on the cumulative histogram of the step edge magnitude image. Since we are interested in larger magnitudes, in our approach the top $\eta\%$ ($\eta = 3.5$) pixels with the largest magnitudes are selected as edge points. We can easily determine the threshold by investigating the cumulative histogram. The thresholded binary image is shown in Figure 3.6(c).

Edge Thinning and Connected Component Labeling

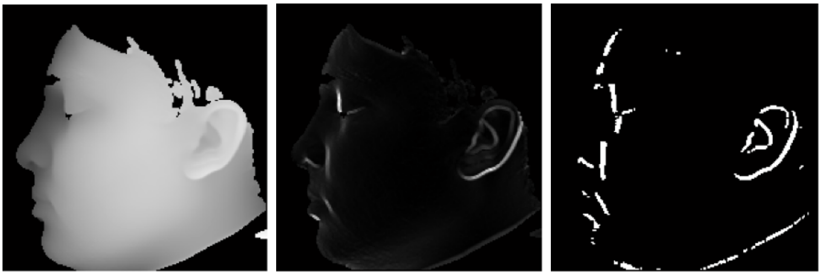
Since some step edge segments are broken, we dilate the binary image to fill the gaps. The dilated image is shown in Figure 3.7(a). We proceed to do edge thinning, and the resulting image is shown in Figure 3.7(b). The edge segments are labeled by running connected component labeling algorithm and some small edge segments (less than 10 pixels) are removed. The left over edge segments are shown in Figure 3.7(c).

Clustering Edge Segments

After edge segments are extracted, those close to each other are grouped into clusters. The clustering procedure works as follows:

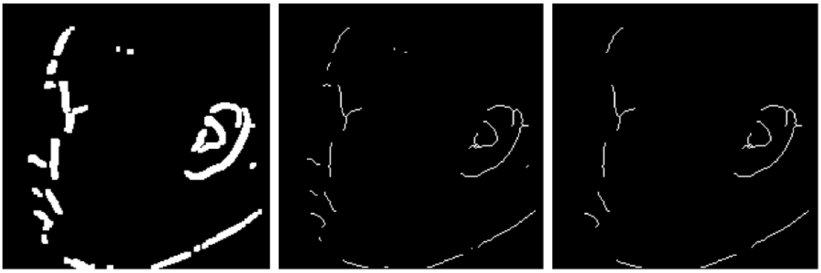
while the number of edge segments > 0

- $i = 0$
- Put the first edge segment e_i into a cluster C_i, and calculate its centroid $\{\mu_{xi}, \mu_{yi}\}$
- For all the other edge segments e_j
  - Calculate the centroid $\{\mu_{xj}, \mu_{yj}\}$
  - if $\max\{|\mu_{xj} - \mu_{xi}|, |\mu_{yj} - \mu_{yi}|\} \leq \epsilon$ put e_j into the cluster C_i, remove e_j and update the cluster's centroid.
- $i = i + 1$ and relabel the edge segments.



(a) (b) (c)

Fig. 3.6. (a) Original side face range image. (b) Step edge magnitude image. (c) Step edge image after thresholding.



(a) (b) (b)

Fig. 3.7. (a) Dilated edge image. (b) Thinned edge image. (c) Left over edge segments.

Three examples of clustering results are shown in Figure 3.8. The first row of Figure 3.8 shows side face range images. The second row shows the corresponding clustering results where each cluster is bounded by a red rectangular box.

Locating Ears by Use of the Ear Shape Model

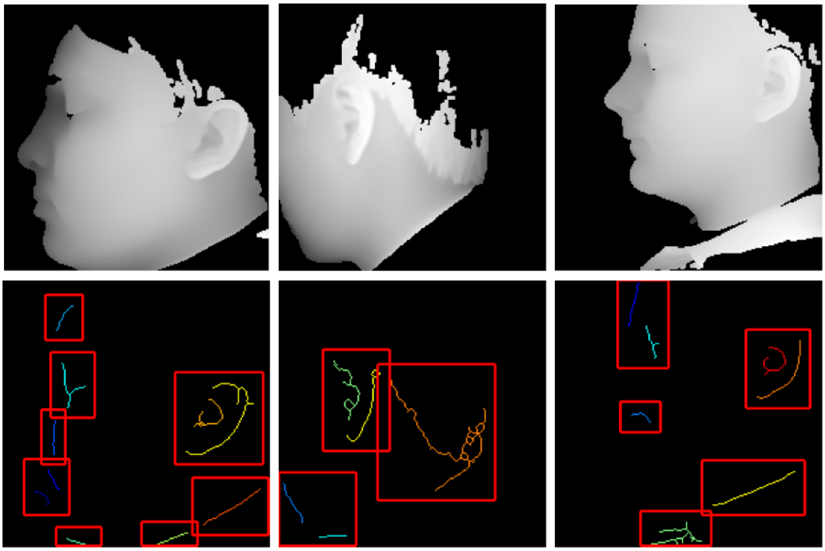
For each cluster obtained in the previous step, the problem of locating ears is to minimize the mean square error between the ear shape model vertices and their corresponding edge vertices in each cluster,

$$E = \frac{1}{n} \sum_{l=1}^{n} |T_r(s_i) - V(s_i)|^2 \tag{3.15}$$

where T_r is the rigid transformation and $V(s_i)$ is a vertex in the 3D side face image closest to the $T_r(s_i)$. The iterative closest point (ICP) algorithm developed by Besl and Mckay [63] is a well-known method to align 3D shapes. ICP requires that each point in one set has a corresponding point in the other set. However, one cannot guarantee that edge vertices in the potential regions satisfy this requirement. Therefore, we use a modified ICP algorithm presented by Turk [64] to register the ear shape model with the edge vertices. The steps of modified ICP algorithm to register a test shape Y to a model shape X are:

1. Initialize the rotation matrix R_0 and translation vector T_0.
2. Find the closest point in X for each given point in Y.
3. Discard pairs of points which are too far apart.
4. Find the rigid transformation (R, T) such that E is minimized.
5. Apply the transformation (R, T) to Y.
6. Go to step 2 until the difference $|E_k - E_{k-1}|$ in two successive steps falls below a threshold or the maximum number of iterations is reached.

By initializing the rotation matrix R_0 and translation vector T_0 to the identity matrix and difference of centroids of two vertex sets respectively, we run ICP iteratively and finally get the rotation matrix R and translation vector T, which brings the ear shape model vertices and edge vertices into alignment. The cluster with minimum mean square error is declared as the detected ear region; the ear helix and anti-helix parts are identified in this process.



10 segments
8 clusters

5 segments
3 clusters

7 segments
5 clusters

Fig. 3.8. Examples of edge clustering results using only range images.

3.1.3 Experimental Results

Data

While we were investigating the above two approaches for 3D ear detection, we had a dataset of 52 subjects with 312 images. Each subject has at least four images. All the experimental results reported in this subsection are on the 52-subject dataset.

Results for the Template Matching Approach

We test the template matching based detection method on 312 side face range images. The parameters of the approach are $\alpha = 0.35$, $w = 5$ pixels, $\gamma = 35$ pixels and $\beta = 99$ pixels. The bin size of the histogram is 0.02. Figure 3.9 shows examples of positive detection in which the detected ears are bounded by rectangular boxes. If the detected region contains a part of an ear, we consider it a positive detection; otherwise it is a false detection. From Figure 3.9, we observe that the ear region is correctly detected. However we may obtain a part of an ear; also we may obtain parts that do not belong to an ear. Figure 3.10 shows examples of false detection. Each column in this figure shows the step edge magnitude image, the dilated binary edge map and the detection result, respectively. We have false detections since the ear helix part is not extracted. The average time to detect an ear from a side face range image is 5.2 seconds with Matlab implementation on a 2.4G Celeron CPU. We achieve a 92.4% detection rate.

Results for the Shape Model Based Approach

We test the ear shape model based detection method on 312 side face range images. If the ear shape model is aligned with the ear helix and anti-helix parts, we classify it as a positive detection; otherwise it is a false detection. In our experiments, the number of vertices in the ear shape model is 113; the average number of edge segments is 6; and the average number of clusters is 4. The average time to detect an ear from a side face range image is 6.5 seconds with Matlab implementation on a 2.4G Celeron CPU. Examples of positive detection results are shown in Figure 3.11. In Figure 3.11, the transformed ear shape model marked by yellow points is superimposed on the corresponding textured 3D face. From Figure 3.11, we can observe that the ear is

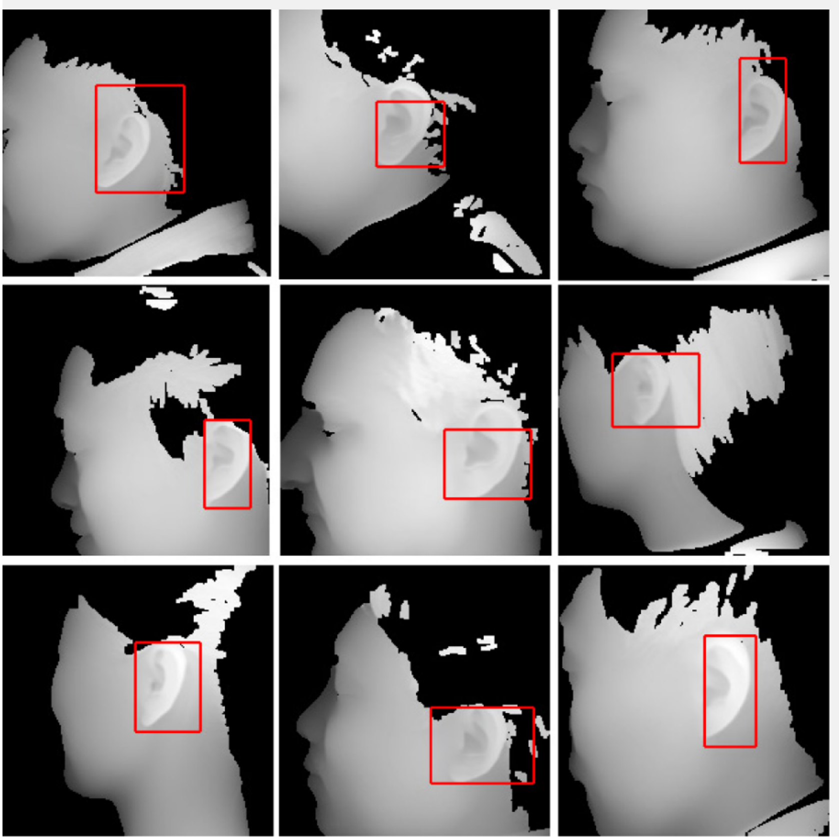


Fig. 3.9. Examples of positive detection using the template matching approach.

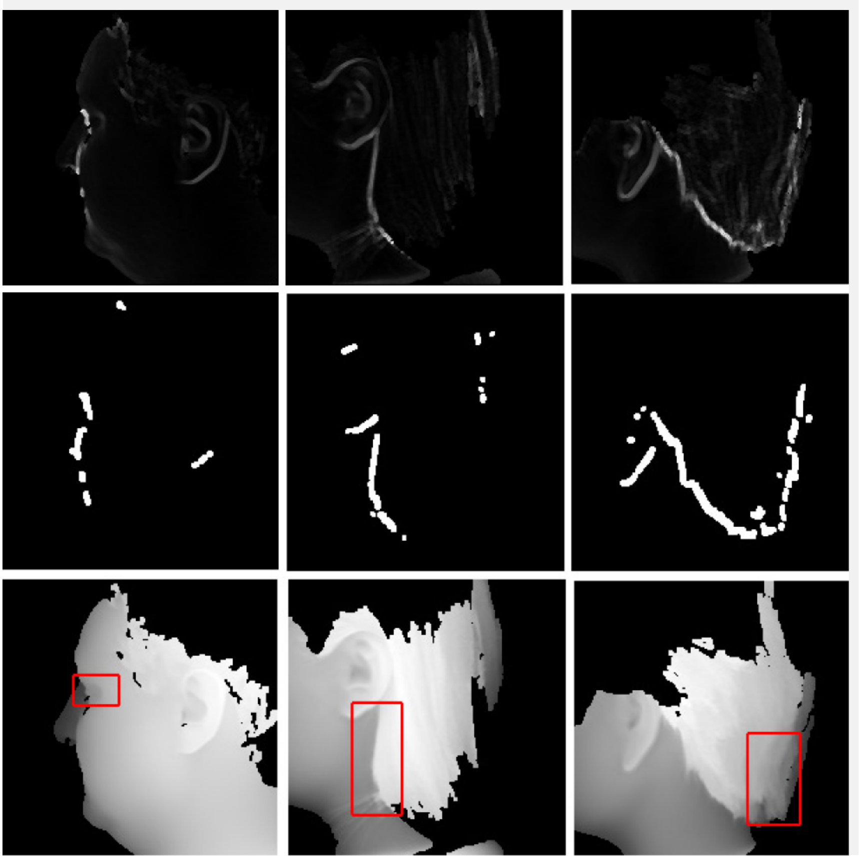


Fig. 3.10. Examples of false detection using the template matching approach. Each column shows the step edge magnitude image, the dilated binary edge map and the detection result, respectively.

correctly detected and the ear helix and anti-helix parts are identified from side face range images. The distribution of mean square error defined in equation (3.15) for the positive detection is shown in Figure 3.12. The mean of mean square error is 1.79 mm. We achieve a 92.6% detection rate.

After we locate the ear helix and anti-helix parts in a side face range image, we put a minimum rectangular bounding box that contains the detected ear. Figure 3.13 shows the examples of detected ears with a red bounding box. From Figure 3.13 and Figure 3.9, we observe that the ears in side face range images are more accurately located by the shape model-based approach. For the failed cases, we notice that there are some edge segments around the ear region caused by hair, which bring more false edge segments. This results in clusters that cannot include the ear helix and anti-helix parts. Since the ICP algorithm cannot converge due to the existence of outliers, the false detection happens; these cases are shown in Figure 3.14 and Figure 3.15. The original face range images and the corresponding edge clusters are shown in Figure 3.14. In this figure, the first row shows face images; the second row shows edge clustering results. The textured 3D faces with overlaid detected ear helix and anti-helix are shown in Figure 3.15.

3.2 3D Ear Detection Using Range and Color Images

In the above two approaches, there are some edge segments caused by non-skin pixels, which result in the false detection. Since the Minolta range sensor provides a registered 3D range image and a 2D color image, we can achieve a better detection performance by fusion of the color and range images.

The flow chart for the fusion of range and color images and global-to-local registration based detection is shown in Figure 3.16. We propose a two-step approach using the registered 2D color and range images by locating the ear helix and the anti-helix parts [1].

In the first step a skin color classifier is used to isolate the side face in an image by modeling the skin color and non-skin color distributions as a mixture of Gaussians [65]. The edges from the 2D color image are combined with the step edges from the range image to locate regions-of-interest (ROIs) that may contain an ear.

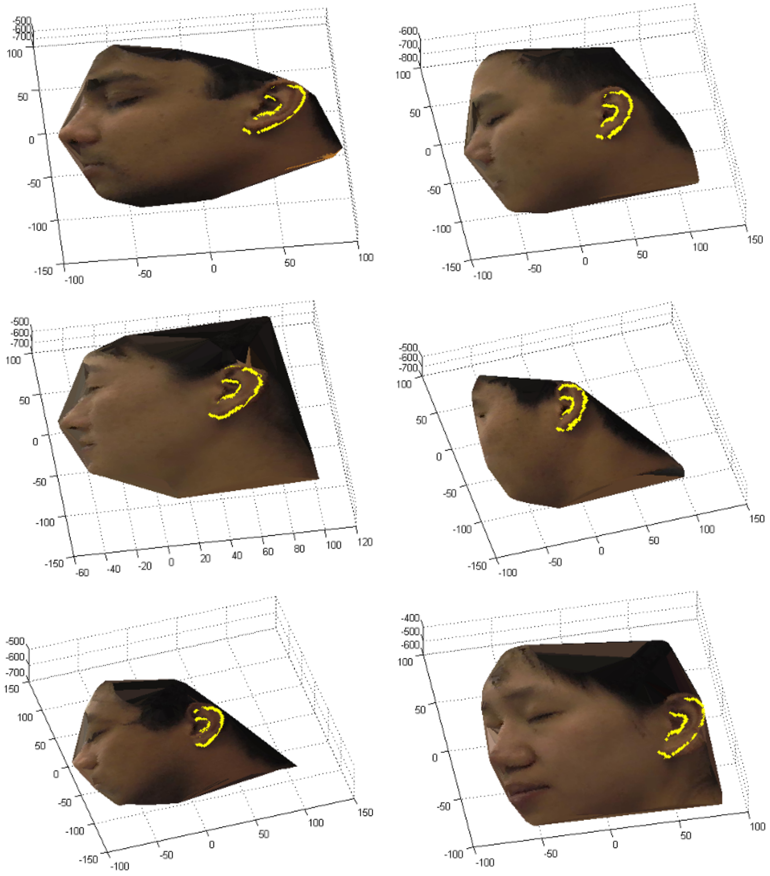


Fig. 3.11. Examples of positive detection results using the shape model based approach.

In the second step, to locate an ear accurately, the reference 3D ear shape model, which is represented by a set of discrete 3D vertices on the ear helix and the anti-helix parts, is adapted to individual ear images by following a new global-to-local registration procedure instead of training an active shape model [66] built from a large set of ears to learn the shape variation.

The DARCES (data-aligned rigidity-constrained exhaustive search) algorithm [67], which can solve the 3D rigid registration problem efficiently and reliably, without any initial estimation, is used to perform

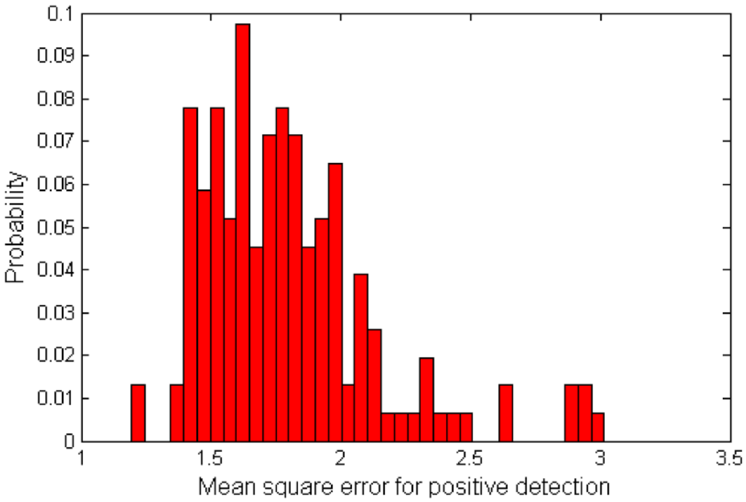


Fig. 3.12. Distribution of the mean square error for positive detection using the shape model based approach.

the global registration. This is followed by the local deformation process where it is necessary to preserve the structure of the reference ear shape model since neighboring points cannot move independently under the deformation due to physical constraints. The bending energy of thin plate spline [68], a quantitative measure for non-rigid deformations, is incorporated into the proposed optimization formulation as a regularization term to preserve the topology of the ear shape model under the shape deformation. The optimization procedure drives the initial global registration towards the ear helix and the anti-helix parts, which results in the one-to-one correspondence of the ear helix and the anti-helix between the reference ear shape model and the input image.

3.2.1 Regions-of-Interest (ROIs) Extraction

Since the images in two modalities (range and color) are registered, the ROIs can be localized in any one modality if they are known in the other modality.

• Processing of Color Images

The processing consists of two major tasks.
► *Skin Color Classification*: Skin color is a powerful cue for segmenting the exposed parts of the human body. Jones and Rehg [65] built a

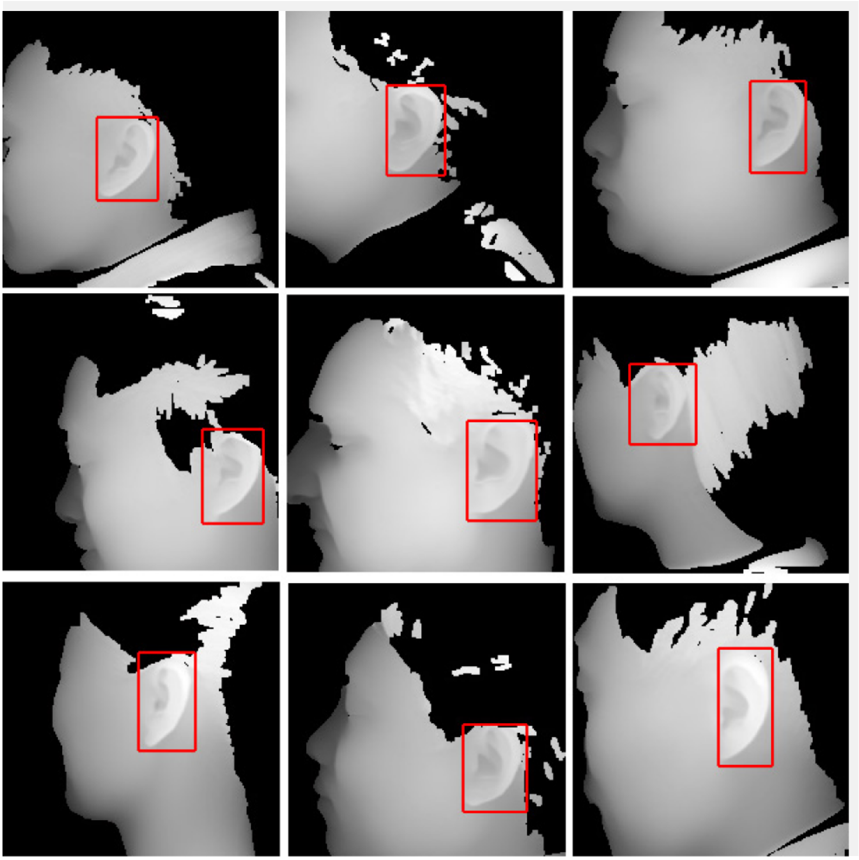


Fig. 3.13. Examples of positive detection using the shape model based approach.

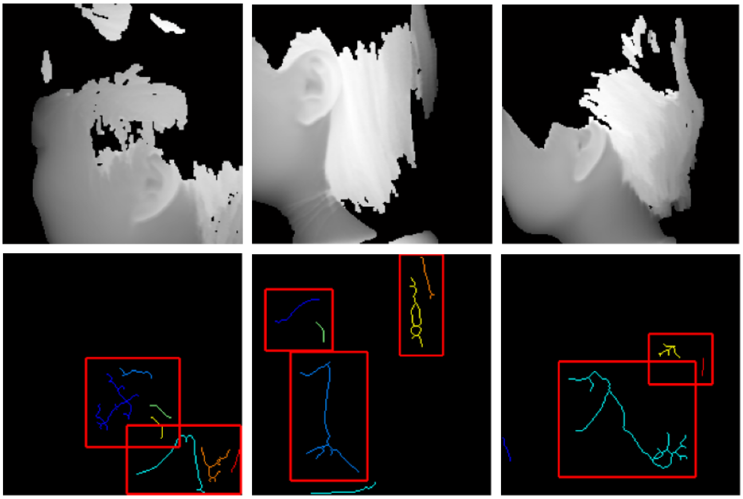


Fig. 3.14. Examples of failed cases using the shape model based approach. Each column shows the range image and the edge clustering result, respectively.

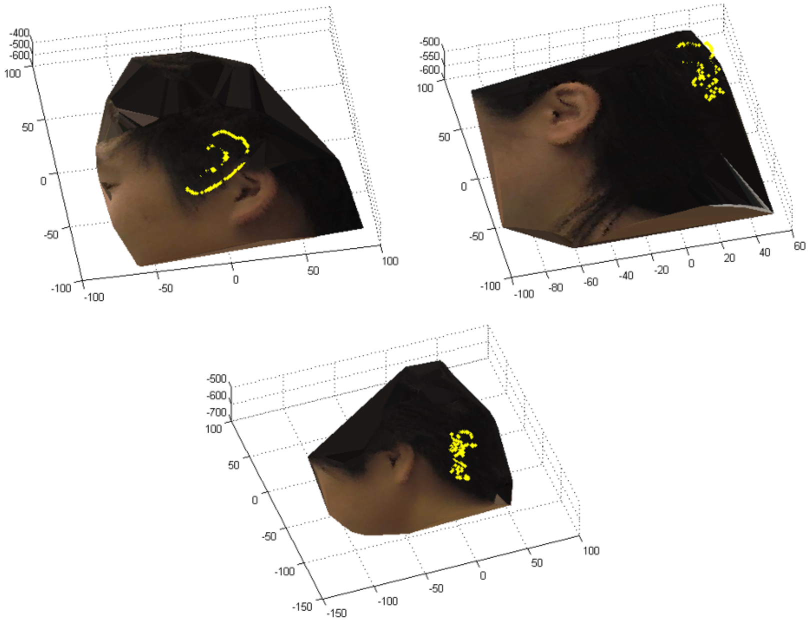


Fig. 3.15. Examples of false detection results using the shape model based approach.

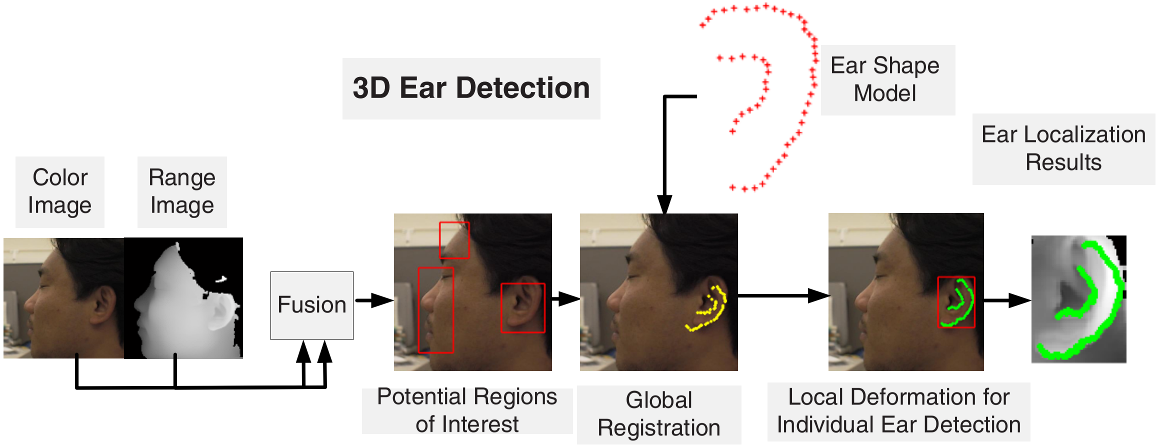


Fig. 3.16. The flow chart of ear detection by a fusion of color and range images.

classifier by learning the distributions of skin and non-skin pixels from a dataset of nearly 1 billion labeled pixels. The distributions are modeled as a mixture of Gaussians and their parameters are given in [65]. We use this method for finding skin regions. When a pixel $p(R, G, B)$ is presented for the classification, we compute *a posteriori* probability $P(\text{skin/RGB})$ and $P(\text{non-skin/RGB})$ and make the classification using the Bayesian decision theory. Figure 3.17(a) shows a color image and Figure 3.17(b) shows the pixel classification result in which the skin pixels are shown as white. We observe that the large skin region containing the ear is roughly segmented.

► *Edge Extraction in Intensity Images*: There are edges, around the ear helix and anti-helix parts, caused by a change in intensity. These are helpful for locating the ear region. The edges are extracted from 2D intensity images. The (R, G, B) color images are first converted to the grayscale images (eliminating the hue and saturation information while retaining the luminance) and then edges are extracted by using the Laplacian of Gaussian (LOG) edge detector (13×13 window is used). Figure 3.17(c) shows the edge detection result using the LOG detector.

• Processing of Range Images

As described in Section 3.1.2, we compute a step edge magnitude image for a given side face range image. Figure 3.18(a) shows a range image in which the darker pixels are far away from the camera; Figure 3.18(b) shows the step edge magnitude image in which the pixels with larger magnitudes are displayed as brighter pixels. We observe that the edge magnitude is large around the ear helix and the anti-helix parts.

Fusion of Color and Range Images

This involves the following steps.

1. The range sensor provides a range mask indicating *valid pixels* (in white), which is shown in Figure 3.19(a).
2. The range mask is combined with the skin color map to generate a final mask indicating the *valid skin pixels*, which is shown Figure 3.19(b).

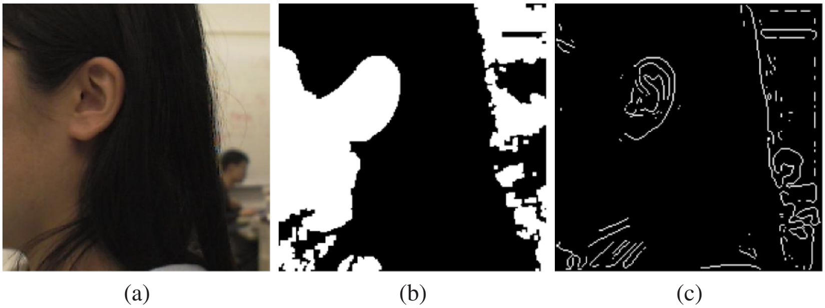


Fig. 3.17. One example of processing the color image. (a) Color image. (b) Skin color map. (c) Edge detection using a LOG edge detector.

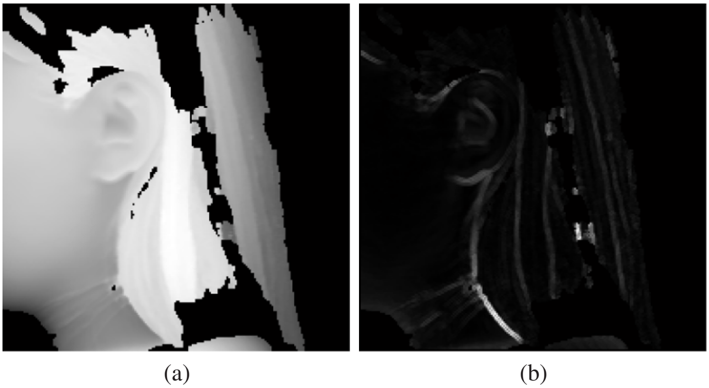


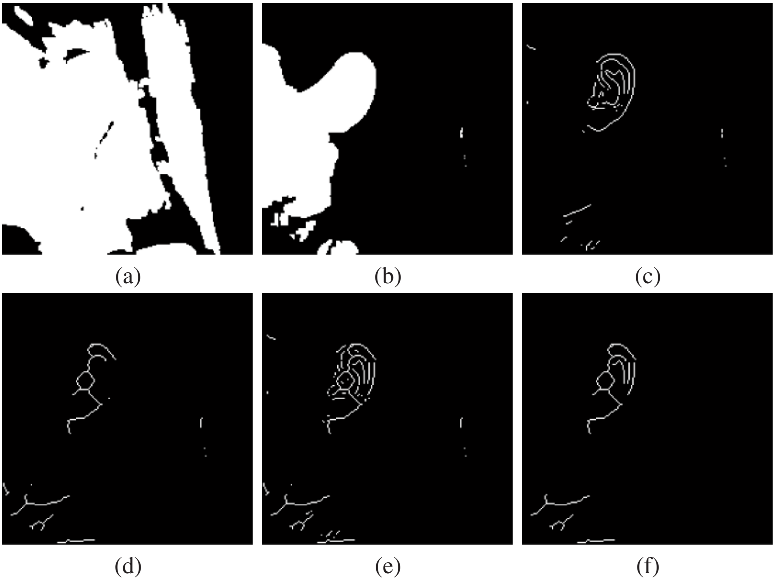
Fig. 3.18. One example of processing the range image. (a) Range image. (b) Step edge magnitude image. In image (a), the darker pixels are away from the camera and the lighter ones are closer. In image (b), the bright pixels denote large edge magnitude.

3. The final mask is applied to edge pixels from the intensity image to remove some of the pixels which are *non-skin* pixels or *invalid* pixels. “Non-skin pixels” mean the pixels that are not on the skin. “Invalid pixels” mean that the range sensor did not make measurements for these pixels. The edge pixels that are left over are shown in Figure 3.19(c).
4. For the range image, the final mask is also applied to the step edge magnitude image. In order to get edges in the range image, the step edge magnitude image is thresholded. The selection of the threshold value is based on the cumulative histogram of the step edge magnitude image. Since we are interested in larger magnitudes, the top $\eta\%$ ($\eta = 3.5$) pixels with the largest magnitudes are selected as the edge pixels. The thresholded binary image is then dilated (using a 3×3 square structuring element) and thinned (shrinking to a minimally connected stroke). The edges so obtained are shown in Figure 3.19(d).
5. The edges from the intensity image and range images are combined in the following manner. The final edge map that we expect to obtain is initialized to be the edge map of the range image (Figure 3.19(d)); for each edge pixel in the intensity image (Figure 3.19(c)) if none of its neighbors are edge pixels in the range image, then this edge pixel is added to the final edge map. An example of the final edge map is shown in Figure 3.19(e).
6. The edge pixels are labeled by the connected component labeling algorithm and the small edge segments are removed (less than 10 pixels in our experiments). The final left over edge segments are shown in Figure 3.19(f).

- **Clustering Edge Segments**

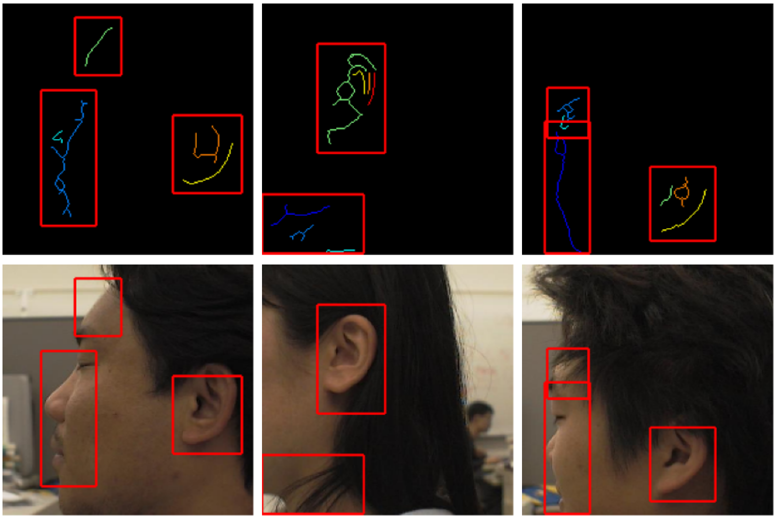
After edge segments are extracted, those close to each other are grouped into clusters. Each cluster is a region-of-interest. The clustering procedure works as described in Section 3.1.2.

Three examples of clustering results are shown in the first row of Figure 3.20, in which each cluster is bounded by a rectangular box and each edge segment is shown in different color. The second row shows the extracted regions-of-interest bounded by boxes overlaid on the color images. From Figure 3.20, we observe that the ear region is correctly identified.


(a) (b) (c)

(d) (e) (f)

Fig. 3.19. Fusion of 2D color and 3D range images. (a) The range mask. (b) The final mask obtained by a combination of the range mask and the skin color map. (c) Edges in the intensity image after applying the final mask. (d) Edges in the range image after applying the final mask. (e) Combination of edges in both color and range images. (f) Edges after removal of small edge segments.



3 clusters 2 clusters 3 clusters

Fig. 3.20. Examples of edge clustering results using the range and color images. The edge segments are shown in the first row where each cluster is bounded by a rectangular box. The second rows show the ROIs superimposed on the color images.

3.2.2 Reference Ear Shape Model

Instead of training an active shape model to learn the shape variation, we adapt the reference ear shape model to input images by following a global-to-local procedure described below, in which the topology of the ear shape model is preserved during the shape deformation. We build the reference ear shape model from an instance of an ear belonging to a person which is described in Section 3.1.2.

3.2.3 Alignment of the Reference Ear Shape Model with a Region-of-Interest

Once a ROI is extracted, the ear helix and the anti-helix parts are identified by the alignment of ROI with the ear shape model. Since the rigid registration cannot account for the local shape variation between ears, we develop a global-to-local procedure: the global registration brings the reference ear shape model into coarse alignment with the ear helix and the anti-helix parts; the local deformation driven by the optimization formulation (given below) drives the reference ear shape model more close to the ear helix and the anti-helix parts.

• Global Rigid Registration

For 3D registration problem, the iterative closest point (ICP) algorithm [63] is widely used for matching points with unknown corresponding pairs. Although there are many variants of the ICP algorithm [69–71], basically it consists of two iterative steps:

1. identifying correspondences by finding the closest points;
2. computing the rigid transformation based on the corresponding pairs.

The major drawback of an ICP-based algorithm is that it needs a good initial guess of the true transformation.

The RANSAC-based data-aligned rigidity-constrained exhaustive search algorithm (DARCES) [67] can solve the registration problem without any initial estimation by using rigidity constraints to find the corresponding points. First three points (primary, secondary, and auxiliary) in the reference surface are selected; then each point on the test surface is assumed to be in correspondence to the primary point, and the other two corresponding points are found based on the rigidity constraints. For every corresponding triangle, a rigid transformation is computed and the transformation with the maximum number of overlapping points is chosen as the solution. Due to its exhaustive nature of the search, the solution it finds is the true one.

In our case, the 3D coordinates of the reference ear shape model are known. We use the DARCES algorithm to find the corresponding triangles (between the reference ear shape model and the ROI under consideration) and the initial transformation. The ICP algorithm is then used to refine the transformation. This process is repeated for each ROI and the ROI with the minimum registration error is passed to the local deformation stage.

- **Local Deformation**

(i) **Thin Plate Spline Transformation:** The reference shape model (the ear helix and the anti-helix parts) is deformed after it is globally aligned with a ROI. Thin plate spline (TPS) transformation is a powerful tool for modeling the shape deformation and is widely used in shape matching [68, 72–74]. The TPS $\mathcal{R}^2 \rightarrow \mathcal{R}^2$ mapping function is defined by the following equation:

$$\mathbf{v} = f(\mathbf{u}) = \begin{bmatrix} f^x(\mathbf{u}) \\ f^y(\mathbf{u}) \end{bmatrix} = A\mathbf{u} + \mathbf{t} + \sum_{i=1}^{n} \begin{bmatrix} w_i^x \\ w_i^y \end{bmatrix} \phi(|\mathbf{u} - \mathbf{u}_i|) \quad (3.16)$$

where $\phi(r) = r^2 \log r$, $\mathbf{u} = [\hat{x}, \hat{y}]^T$, $\mathbf{v} = [x, y]^T$ and A and $\mathbf{t}$ form an affine transformation given by

$$[A \;\; \mathbf{t}] = \begin{bmatrix} a_{00} & a_{01} & t_0 \\ a_{10} & a_{11} & t_1 \end{bmatrix}.$$

The $n \times 2$ matrix W is given by

$$\begin{bmatrix} w_1^x & w_2^x & \cdots & w_n^x \\ w_1^y & w_2^y & \cdots & w_n^y \end{bmatrix}^T.$$

It specifies the non-linear warping where n is the number of landmark points. Given n landmark points $\mathbf{u}(\hat{x}_i, \hat{y}_i)$ and their corresponding points $\mathbf{v}(x_i, y_i)$, equation (3.16) can be rewritten as $2n$ linear equations. However there are $2n+6$ unknown parameters to be solved. The following six constraints are added to make the spline function (equation (3.16)) have the square integrable second derivatives:

$$P^T[w_1^x, w_2^x, \cdots, w_n^x]^T = 0, \;\; P^T[w_1^y, w_2^y, \cdots, w_n^y]^T = 0 \quad (3.17)$$

where P is a $n \times 3$ matrix defined by $(\mathbf{1}, \hat{\mathbf{x}}, \hat{\mathbf{y}})$, $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_n)^T$ and $\hat{\mathbf{y}} = (\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_n)^T$. The $2n + 6$ equations can be put into a compact matrix form:

$$\begin{bmatrix} \Phi & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} W \\ \mathbf{t}^T \\ A^T \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{0} \end{bmatrix} \quad (3.18)$$

where the $n \times n$ matrix $\Phi_{ij} = \phi(\mathbf{u}_i - \mathbf{u}_j)$, $\mathbf{v} = (\mathbf{x}, \mathbf{y})$, $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ and $\mathbf{y} = (y_1, y_2, \cdots, y_n)^T$. The TPS transformation minimizes the following bending energy function,

$$B_e = \iint_{\mathcal{R}^2} (F(f^x) + F(f^y))dxdy \quad (3.19)$$

where $F(g^*(x, y)) = (g_{xx}^2 + 2g_{xy}^2 + g_{yy}^2)^*$, $*$ denotes the $(x$ or $y)$ under consideration and g_{xx}, g_{xy} and g_{yy} are second order derivatives. It can be shown that the value of bending energy is $B_e = \frac{1}{8\pi}(\mathbf{x}^T K\mathbf{x} + \mathbf{y}^T K\mathbf{y})$ where $\mathbf{x} = (x_1, x_2, \cdots, x_n)^T$ and $\mathbf{y} = (y_1, y_2, \cdots, y_n)^T$ [68]. The matrix K is the $n \times n$ upper left matrix of

$$\begin{bmatrix} \Phi & P \\ P^T & 0 \end{bmatrix}^{-1},$$

which only depends on the coordinates of the landmark points in $\{\mathbf{u}\}$. Therefore, the bending energy is determined by the coordinates of landmark points and their correspondences. Furthermore, the bending energy is a good measurement of the shape deformation. Since the coordinates of the reference ear shape model are known, the matrix K can be precomputed. The task is to drive the reference ear shape model towards the ROI ear such that the topology of the reference ear shape model is preserved. The bending energy is used to penalize the large shape deformation.

(ii) **Optimization Formulation:** In Section 3.2.1, we noted that there are strong step edge magnitudes in range images around the ear helix and the anti-helix parts. After we bring the reference shape model into coarse alignment with the ear helix and the anti-helix parts (in the ROI image) through the global rigid registration, we get the locations of the 3D coordinates of ear helix and anti-helix parts in the 2D color image and perform the local deformation on the 2D image plane since the 2D color image is registered with the 3D range image. In other words, we would like to drive the reference ear shape model more close to the ear helix and the anti-helix parts with the topology of the shape model preserved. We can achieve this task by minimizing the proposed new cost function:

$$\begin{aligned} E(\mathbf{x},\mathbf{y}) &= E_{img}(\mathbf{x},\mathbf{y}) + \gamma E_D(\mathbf{x},\mathbf{y}) \\ &= \sum_{i=1}^{n} h(|\nabla I_{\text{step}}(x_i,y_i)|) \\ &\quad + \frac{1}{2}\gamma(\mathbf{x}^T K\mathbf{x} + \mathbf{y}^T K\mathbf{y}) \end{aligned} \tag{3.20}$$

where $h(|\nabla I_{\text{step}}|) = 1/(1 + |\nabla I_{\text{step}}|)$, $|\nabla I_{\text{step}}(x_i,y_i)|$ is the step edge magnitude of ith point of the shape model located in the 2D plane and γ is a positive regularization constant that controls the topology of the shape model. For example, increasing the magnitude of γ tends to keep the topology of the ear shape model unchanged. In equation (3.20), the step edge magnitude in range images is used for the term E_{img} since edges in range images are less sensitive to the change of viewpoint and illumination than those in color images. In equation (3.20) the first term E_{img} drives points $(\mathbf{x},\mathbf{y})$ towards the ear helix and the anti-helix parts which have larger step edge magnitudes; the second term E_D is the bending energy that preserves the topology of the reference shape model under the shape deformation. When we take the partial derivatives of equation (3.20) with respect to $\mathbf{x}$ and $\mathbf{y}$ and set them to zero, we have

$$\begin{aligned} \gamma K\mathbf{x} - \sum_{i=1}^{n} \frac{1}{(1 + |\nabla I_{\text{step}}(x_i,y_i)|)^2} \varOmega^{\mathbf{x}} = 0, \\ \gamma K\mathbf{y} - \sum_{i=1}^{n} \frac{1}{(1 + |\nabla I_{\text{step}}(x_i,y_i)|)^2} \varOmega^{\mathbf{y}} = 0. \end{aligned} \tag{3.21}$$

In equation (3.21),

$$\begin{aligned} \varOmega^{\mathbf{x}} &= \frac{\partial |\nabla I_{\text{step}}(x_i,y_i)|}{\partial \mathbf{x}} \\ \varOmega^{\mathbf{y}} &= \frac{\partial |\nabla I_{\text{step}}(x_i,y_i)|}{\partial \mathbf{y}}. \end{aligned}$$

Since K is positive semidefinite, equation (3.21) can be solved *iteratively* by introducing a step size parameter α which is shown in equation (3.22) [75]. The solutions can be obtained by matrix inversion which is shown in equation (3.23), where I is the identity matrix.

$$\begin{aligned} \gamma K \mathbf{x}_t + \alpha(\mathbf{x}_t - \mathbf{x}_{t-1}) - \mathcal{F}^{\mathbf{x}}_{t-1} = 0 \\ \gamma K \mathbf{y}_t + \alpha(\mathbf{y}_t - \mathbf{y}_{t-1}) - \mathcal{F}^{\mathbf{y}}_{t-1} = 0 \end{aligned} \tag{3.22}$$

In equations (3.22) and (3.23),

$$\begin{aligned} \mathcal{F}^{\mathbf{x}}_{t-1} &= \sum_{i=1}^{n} \frac{1}{(1 + |\nabla I^{t-1}_{step}(x_i, y_i)|)^2} \frac{\partial |\nabla I^{t-1}_{step}(x_i, y_i)|}{\partial \mathbf{x}} \\ \mathcal{F}^{\mathbf{y}}_{t-1} &= \sum_{i=1}^{n} \frac{1}{(1 + |\nabla I^{t-1}_{step}(x_i, y_i)|)^2} \frac{\partial |\nabla I^{t-1}_{step}(x_i, y_i)|}{\partial \mathbf{y}}. \end{aligned}$$

$\mathcal{F}^{\mathbf{x}}_{t-1}$ and $\mathcal{F}^{\mathbf{y}}_{t-1}$ are evaluated for coordinates (x_i, y_i) at the iteration $t-1$. $|\nabla I^{t-1}_{step}(x_i, y_i)|$ is the step edge magnitude at the location of (x_i, y_i) at the iteration $t-1$. We have used $\alpha = 0.5$ and $\gamma = 100$ in our experiments.

$$\begin{aligned} \mathbf{x}_t &= (\gamma K + \alpha I)^{-1} \Big(\alpha \mathbf{x}_{t-1} + \mathcal{F}^{\mathbf{x}}_{t-1} \Big) \\ \mathbf{y}_t &= (\gamma K + \alpha I)^{-1} \Big(\alpha \mathbf{y}_{t-1} + \mathcal{F}^{\mathbf{y}}_{t-1} \Big) \end{aligned} \tag{3.23}$$

3.2.4 Experimental Results

The detection experiments are performed on the UCR dataset (155 subjects with 902 shots) and the UND dataset Collection F (302 subjects with 302 pairs) and a subset of Collection G (24 subjects with 96 shots).

- **Ear Detection on UCR Dataset**

The proposed automatic ear detection method is tested on 902 pairs of range and color images. Figure 3.21 shows the effectiveness of the global-to-local registration procedure on three people. After the global registration, we get the positions of the 3D coordinates on the 2D image plane and their locations are marked by the bright dots which are

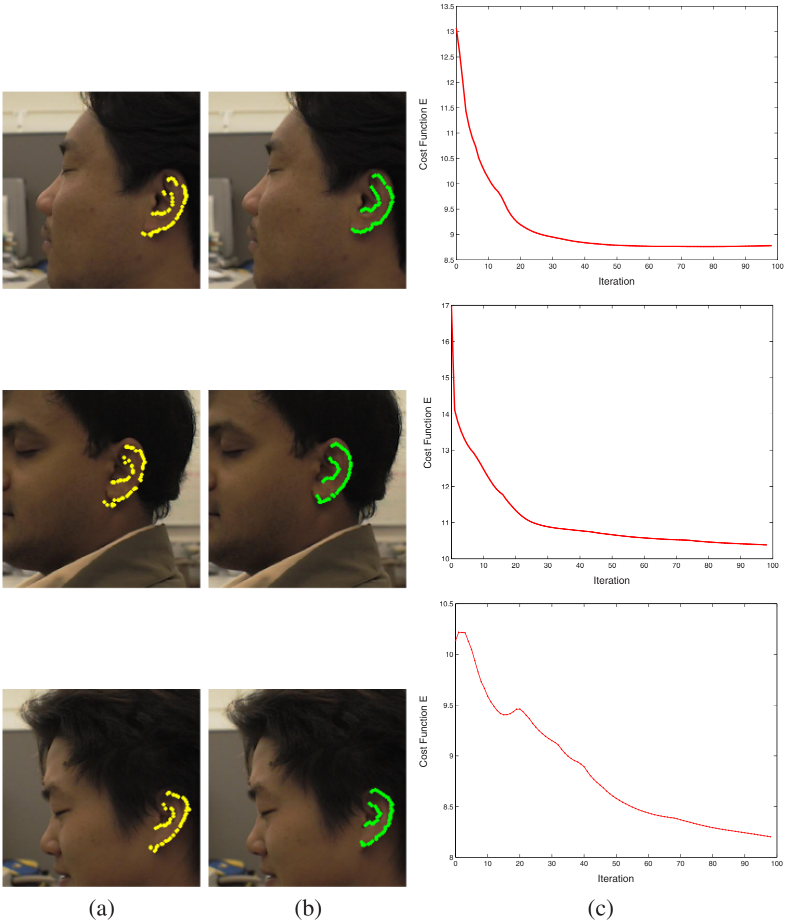


Fig. 3.21. Examples of global registration and local deformation. (a) Global registration results superimposed on the color images. (b) Local deformation results superimposed on the color images. (c) Cost function (equation (3.20)) vs. iteration.

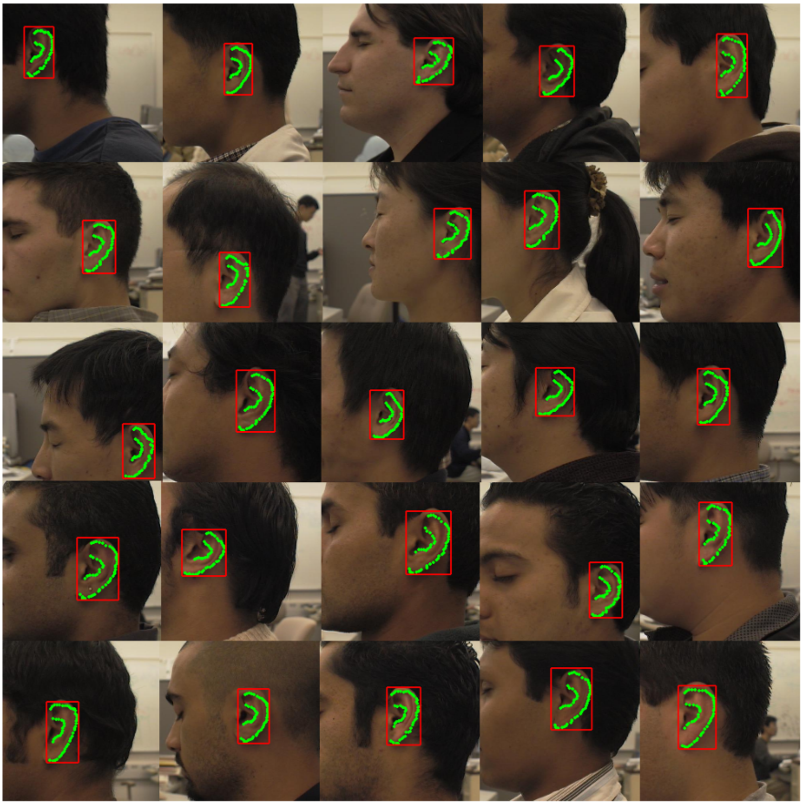


Fig. 3.22. Results of ear localization on the UCR dataset. The helix and the anti-helix parts are marked by the bright dots and the detected ear is bounded by a rectangular box.

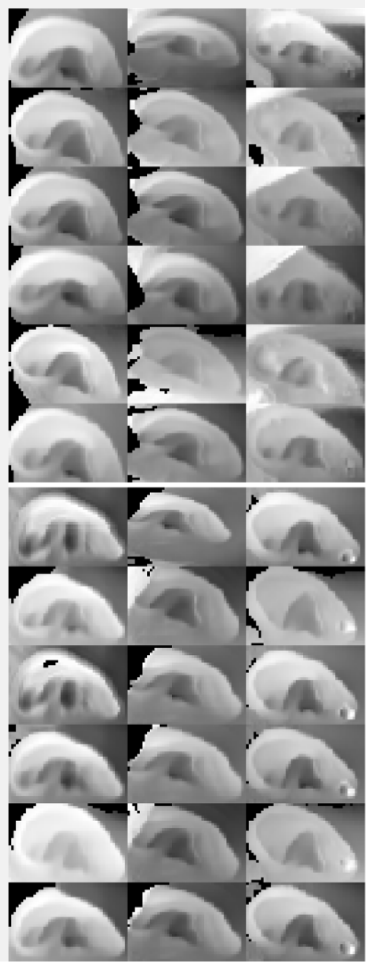


Fig. 3.23. Examples of extracted ears (from left to right and top to bottom) in the side face range images shown in Figure 1.4.

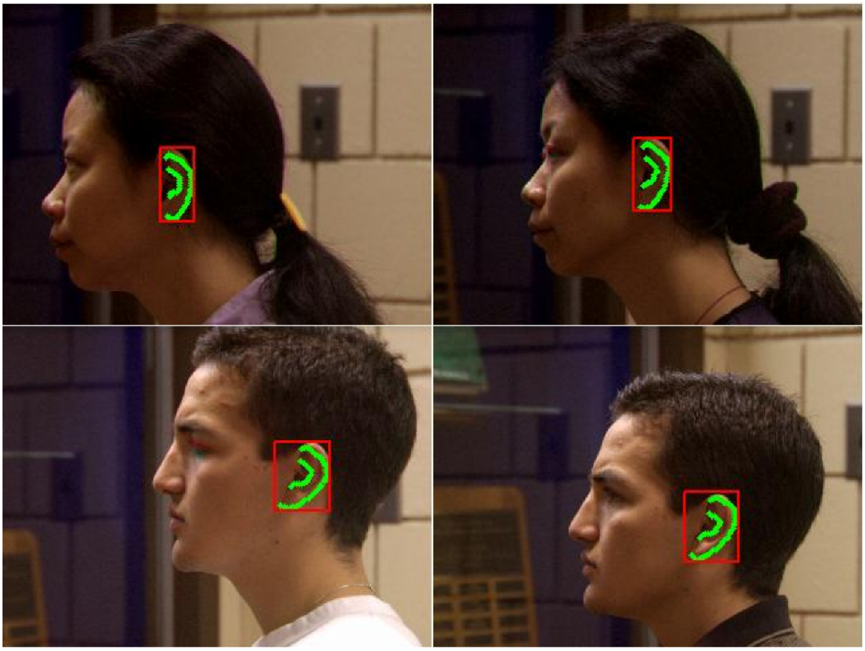


Fig. 3.24. Results of ear localization on the UND dataset shown in Figure 1.5. The helix and the anti-helix parts are marked by the bright dots and the detected ear is bounded by a rectangular box.

Table 3.2. Comparison of the three ear detection approaches.

Ear Detection Method	**Detection Rate**	**Detection Time**
Template matching	92.4% on the UCR dataset (52 subjects with 312 shots)	5.2s
Ear shape model	92.6% on the UCR dataset (52 subjects with 312 shots)	6.5s
Fusion of color/range image & global-to-local registration	99.3% on the UCR dataset (155 subjects with 902 shots), 87.71% on the UND dataset	9.48s

shown in Figure 3.21(a). It can be seen that the shape model is roughly aligned with the ear helix and the anti-helix parts. The ear shape model is then driven towards the ear helix and the anti-helix parts by minimizing the cost function (equation (3.20)) and their locations are marked by the bright dots which are shown in Figure 3.21(b). It can be seen that the optimization formulation drives the shape model more closely to the true positions with the topology of the reference ear shape model preserved. Figure 3.21(c) shows that the cost function decreases with the number of iterations, which means the optimization formulation works. More examples of ear localization are shown in Figure 3.22, in which the detected ear helix and the anti-helix parts are shown by the dots superimposed on the 2D color images and the detected ear is bounded by the rectangular box. We observe that the ears and their helix and anti-helix parts are correctly detected.

In order to quantitatively evaluate the improvement of ear localization through the local deformation driven by the optimization formulation, we compute the error

$$\varepsilon = \frac{1}{N_m} \sum_{i=1}^{N_m} \left(\frac{1}{n} \sum_{j=1}^{n} \mathrm{Dist}(v_{ij}, G_{ti}) \right) \tag{3.24}$$

for the global registration and the local deformation, where N_m is the number of side face range images ($N_m = 208$, since we manually labeled 3D vertices on the ear helix and the anti-helix parts for 208 images for evaluation purposes only), n is the number of points on the shape model, v_{ij} is the jth point on the shape model detected in the ith side face range image, G_{ti} is the set of manually labeled 3D points on the ear helix and the anti-helix parts of the ith side face range image and $Dist(v_{ij}, G_{ti})$ is the distance between v_{ij} and its closest point in

G_{ti}. The error ε for the global registration is 5.4mm; the error ε after the local deformation is 3.7mm. Thus, the local deformation driven by the optimization formulation really improves the localization accuracy. Figure 3.23 shows the extracted ears from the side face range images in Figure 1.4. The average number of points on the ears extracted from 902 side face images is 2,797. The ear detection takes about 9.48s with Matlab implementation on a 2.4G Celeron CPU. If the reference ear shape model is aligned with the ear helix and the anti-helix parts in a side face range image, we classify it as a positive detection; otherwise a false detection. On the 902 side face range images, we achieve 99.3% correct detection rate (896 out of 902).

- **Ear Detection on UND dataset**

Without changing the parameters of the ear detection algorithm on the UCR dataset, the proposed automatic ear detection method is tested on 700 ($302\times2 + 24 \times 4 = 700$) pairs of range and color images of the UND dataset (Collections F and a subset of Collection G). We achieve 87.71% correct detection rate (614 out of 700). The average number of points (on 700 images) on the ears is 6,348. Figure 3.24 shows the extracted ears from the side face range images in which the ear helix and the anti-helix are marked by bright points and the extracted ear is bounded by a rectangular box.

3.3 Conclusions

We have proposed three techniques—template matching based detection, ear shape model based detection, and fusion of color/range images and global-to-local registration based detection—to locate ears from side face range images. The comparison of the three approaches are given in Table 3.2. The first approach runs the fastest and it is simple, effective and easy to implement. The second approach locates an ear more accurately than the first approach since the shape model is used. The third approach uses both color and range images to localize the ear region accurately by following a global-to-local registration procedure. It performs the best on both the UCR and the UND datasets and it runs the slowest. Experimental results on real side face range images demonstrate the effectiveness of the proposed three approaches.