CHAPTER 4

FACIAL FEATURE EXAGGERATION AND CARICATURE GENERATION

In this chapter, facial features will be exaggerated to manifest each people's characteristics. We will present several strategies for feature exaggeration on both plain and stereo, and then adopt image metamorphosis technique to generate various drawing styles of caricatures as the texture map for modeling.

4.1 Average data computation

It is necessary to perform quantitative analysis to obtain important statistics of the face components. To obtain samples for the computation, we generate frontal-view images by interpolating the left and right image pairs, and record the coordinates of all feature points extracted by AAM. In order to compensate for the difference in imaging distance, first we define the origin of the face coordinate system as the midpoint of two master nodes of left and right eye groups (see Fig.4-1), and then normalize all feature nodes' coordinates according to:

$$P(x',y') = \left(\frac{P_x - origin_x}{imageWidth}, \frac{P_y - origin_y}{imageLength}\right) \quad \textbf{(12)}$$

where P(x',y'): normalized coordinates of point p

$origin_x$: x coordinate of the origin

$origin_y$: y coordinate of the origin

P_x: original x coordinate of point *P*

P_y: original y coordintate of point *P*

imageWidth: Width of current image.

imageLength: Length of current image.

Aside from the locations of the individual nodes, the coordinates of the bounding box for each face component are also recorded.

We measure the location, size, and shape of every individual group. The recorded information is categorized into two types:

I. **Global data：**

- Location: recording the average position of each node. The position of the master node is considered as the position of this group and the relative position of each group to its parent group can be determined by measuring the distance between two master nodes.
- Size: recording the size of the groups. The size is defined by the bounding box of the certain group.

II. **Local data：**

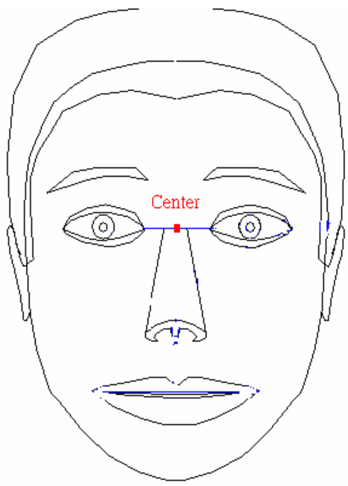


Fig. 4-1: The origin of facial coordinate we define.

- Shape: recording the shape of the group. Regardless of other groups, we normalize each group and measure the local distances between the calibration node and all other nodes in this group. Local data are used to describe the shape of each group.

An ideal exaggeration algorithm should not only preserve the original features of the subject, but also attain the exaggerative effects rather than adopting some pre-defined shape kits. For example, some existing algorithms generate caricatures by compositing parts of facial components drawn by cartoonists [20], while others defined some methods for exaggerating different parts of the same components, such as one for rounding the nose tip and another for elongate the nose bridge [21]. Sometimes works produced by these algorithms may look more like caricaturists' strokes. However, these approaches are clumsy and niggling because a great database has to be maintained. They would miss important types or lose the fidelity of the features if these features are not already present in the system. We simplify those processes by recording the global and local data of groups, and design a strategy to handle all types of facial features in a uniform way. The absolute locations and relative locations between groups can be recovered by comparing with the global data. Distinct shape of each group can be identified by comparing with the local data as well. The following explains how we use the local data to handle the various shapes of groups.

- We empirically select one node for each group as the calibration node, which is the most suited for calibrating all other node in this group. For instance, in the eyebrow group we select the peak of the eyebrow as its calibration node, and the local position of all other nodes in this group are computed according to this calibration node.
- We use the normalization equation

$$\left(X_i', Y_i'\right) = \left(\frac{X_i - C_x}{unitX}, \frac{Y_i - C_y}{unitY}\right) \quad \textbf{(13)}$$

where (C_x ,C_y) is the coordinates of the calibration node and *unitX, unitY* are the width and length of the bounding box, to normalize each group. After that, the average shape of each group can be obtained by averaging the local coordinates of all the nodes in the group. Fig. 4-2 illustrates the computation process graphically.

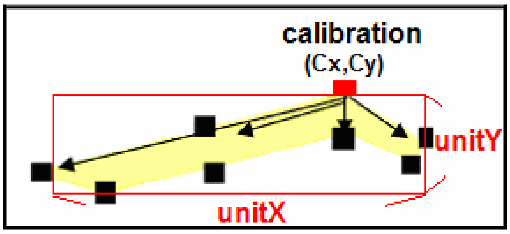


Fig. 4-2: Illustration of the normalization equation

4.2 Facial feature exaggeration

We compare the face mesh of the input image with the average face to determine the exaggeration rate for each component. In a previous implementation reported in [2], shape, size, and location are compared to the mean value separately. The relations among groups have been ignored. However, a group has a 50% probability of being classified as a normal. In the extreme case when all group features fall in the 'normal' range, no exaggeration will be applied. Caricatures thus generated would appear bland. In addition, the exaggeration rate of the linear model is proportional to the difference between the current measurement and the upper/lower bound of the normal range. This also limits the amount of exaggeration. To promote the versatility of the produced caricatures, we propose another strategy which accounts for the relative sizes and positions of different groups of nodes.

Intuitively, when people observe someone's appearance, there is no objective criterion for making a comparison. The judgment is usually based on their own prior experiences and impressions. Therefore, factors having to do with the overall impression are more critical, the most prominent one being the relationship among facial components. Consequently, we adopt different strategies for shape, size, location, and depth exaggeration. For shape and location, if the input component is classified as normal, no shape exaggeration

```
if(value within normal range)
        value not changed
else if(value < range_min)
        value = value – (range_min – value)* scale
else
        value = value + ( value – range_max)* scale
```

Fig. 4-3: Linear model for computing the ratio of exaggeration out of the normal range.

will be applied. Otherwise, a linear mapping shown in Fig. 4-3, is used to obtain the exaggeration rate.

For size exaggeration, we define a percentage value of difference D_i for each group i and compute it according to:

$$d_i = \left| \frac{CurrentSize - MeanSize}{MeanSize} \right| \quad \textbf{(14)}$$

$$D_i = \frac{d_i}{d_{\max}} \quad \textbf{(15)}$$

where CurrentSize and MeanSize are the current size and average size of group *i*, and d_{max} is the maximum of d_i. D_i is a measurement to give a bonus for exaggerative ratio when the exaggeration in size is applied. We divided the exaggeration on size into two steps. In the first stage, we exaggerate the size of facial components by the linear model. In addition, we decide the exaggerative ratio by multiplying D_i and the difference between current size of group *i* and its mean size for another exaggeration. Specifically, exaggeration in size can be expressed as:

$$E_1 = \begin{cases} (Size_{current} - Size_{boundary}) \times scale_I \\ 0 \ \ , if\ Size_{current}\ in\ normal\ range \end{cases} \quad \textbf{(16)}$$

$$E_2 = D_i \times (Size_{current} - Size_{mean}) \times scale_{II} \quad \textbf{(17)}$$

$$Exaggeration\ amount_{size} = E_1 + E_2 \quad \textbf{(18)}$$

Table 4-1 and 4-2 show the statistics of size exaggeration applied to the face database we have collected. Despite E_1 and E_2 could be positive or negative, absolute value is used in all computations as we are concerned with the exaggerative amount. The mean value in Table 4-1 is the average of E_1 from all feature groups expressed in normalized coordinates. Here $scale_I$ is set to 1.05. The displacement amount is about 2.38 pixels in the horizontal (X) direction and 1.72 pixels in the vertical (Y) direction assuming the original image size is 320x240.

Table 4-1: Statistics of E_1

	(X)	(Y)
sum	0.05208	0.02931
mean	0.00744	0.00419
stddev	0.00063	0.00032

Table 4-2: Contribution of E_2 in size exaggeration

average of ratio : $E_2 / (E_1+E_2)$

total number of samples: 107

※G7(face contour) is not employed in E_2 computation

	avg of X ratio	avg of Y ratio	# of $E_1(X) \neq 0$	# of $E_1(Y) \neq 0$
G1	0.7080479	0.6774102	56	57
G2	0.6492418	0.6936893	58	57
G3	0.6070887	0.6338498	55	56
G4	0.5690624	0.6617156	56	57
G5	0.6123052	0.4980849	57	55
G6	0.6700905	0.6772124	56	57
G7	0	0	56	55

We use the same scale to compute E_2 in order to compare the relative contribution of E_1 and E_2. According to Table 4-2, about half of the sample is out of the normal range of each group. In these cases, E_2 contributes more than 50% for the size exaggeration, and the average shift amount of E_2 is about 4.46 in the X direction and 3.14 in the Y direction. In summary, E_2 plays a more important role than E_1 in all cases if the same scaling factor is employed. The two-stage process for implementing the exaggeration effect is thus more prominent than the original linear model.

Finally, for depth information, to manipulate an overall exaggeration on z coordinate is not effective in our work because correlation algorithm is not capable of obtaining precise depth information in other regions on human face. Since prominent facial features are nose and mouth, we assume that exaggerating operation adopted in these two regions is enough to obtain conspicuous effects. Thus, we define several virtual control points (Fig. 4-4) based on feature nodes for the reference, and use the same linear exaggeration model as that of shape and location.

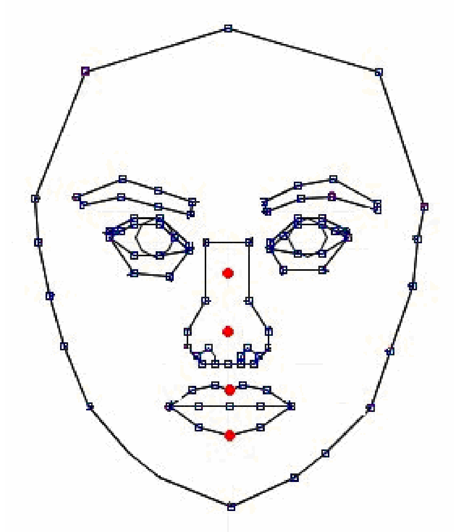


Fig. 4-4: Virtual points (in red) interpolated from feature nodes for the reference of depth

Given the amount of exaggeration for each component, a five-stage process is followed to adjust the positions of the nodes:

1. Initial node placement: At this stage, the positions of the nodes in each group are moved independently. Exaggeration rate is estimated by computing the deviation from normal range. The shift amount is proportional to the exaggeration rate.
2. Size adjustment: The size of the bounding box of the exaggerated shape is used to shrink or expand the component as a whole. Here the average size is employed.
3. Spatial exaggeration: Only left eye, right eye and nose groups are involved at this stage. The criterion is set to be the distance between the origin and the master nodes of each group. The exaggeration rate is also generated based on the linear model and the final position of each master node is determined independently.
4. Final node placement: Spatial relationship among the face components has to be considered to ensure correct placement. This is achieved by placing the nodes

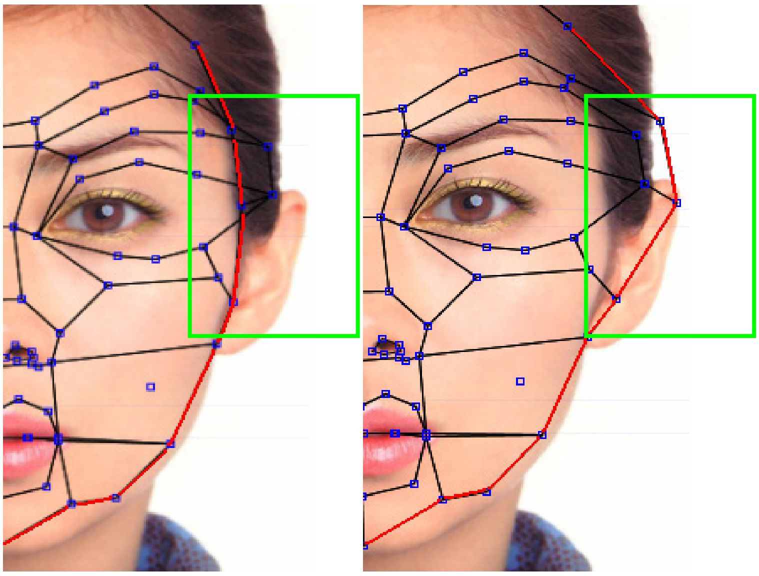
with higher ranks[†] first. The master point of each group is taken as the basis for detecting overlaps or collisions as a result of magnification or excessive shift. Other points in the group shift accordingly with the master node. In our process, nodes in the group with Rank2[*] are firstly relocated, and then the master node of its child groups is placed to maintain the original distance and avoid collision. Finally, all left nodes in lower rank groups are settled. Notice that we are not considering the face contour at this point. It is going to be dealt with in the next stage.

5. Collision Detection: Sometimes collisions between facial features and facial contour may occur after the foregoing processes. For example, the eyes may extend outside the face contour if the subject's eyes are big and the exaggeration rate is too large. To address this issue, we need to move the hierarchy of the face contour to the lowest level when collision takes place. We then check the distances between the specific nodes and adjust the positions of the nodes of group1 (face contour) if the distance is smaller than a minimum value to ensure that all nodes are inside in the face contour. Despite we can partially extend a portion of face contour to avoid collision, as demonstrated in Fig. 4-5, the face contour will be distorted simultaneously.

   A simple method to obtain a more natural looking face contour is by enlargement. Resizing the contour without adjusting the dimension of other facial components, however, will adversely affect the exaggeration rate. In addition, if collision occurs only on one side of the face, uniform expansion of the contour will not eliminate the asymmetry. The two-stage process described below is specifically designed to address these problems.

---

[*] Group hierarchy Rank2: left eye, right eye and nose.



(a) (b)

Fig. 4-5: (a) The exaggerated shape of the eye is out of the face contour when a large exaggeration rate is decided. (b) Partially move the face contour to avoid collision.

- Compute the boundary box including all feature components except face contour and take the intersection of the diagonal lines as the temporary origin.
- Shift the face contour according to the temporary origin, and then compute the enlargement ratio to avoid collision.

This method is able to solve the asymmetry problem we mentioned, but cannot fulfill the compensation of the relative exaggeration rate. Even so, if we think preserving the fidelity and keeping its good look is more important, we will compromise on this imperfection.

Tables 4-3 summarizes the relationship between the scale for relative position exaggeration (size exaggeration is fixed at 0.05) and the number of collisions encountered. In Table 4-4, the exaggeration rate for relative position is set to 2, and the relationship between the size scale and the number of collisions is listed. In both cases, we observe increasing events of collisions as the scale factor becomes larger, which justifies the formulation of the two-stage process for collision detection and eradication.

Table 4-3: Collision as a result of relative position exaggeration

(size scale: 0.05)

scale	# of sample	# of collision	%
0.5 ~ 2	103	18	17.48
2.5 ~ 7	103	19	18.45
7.5 - 8	103	20	19.42
8.5	103	21	20.39
9	103	22	21.36
9.5 ~ 10.5	103	24	23.3
11 ~ 12	103	23	22.33
12.5 ~ 15.5	103	24	23.3
16.5 ~ 17.5	103	25	24.27

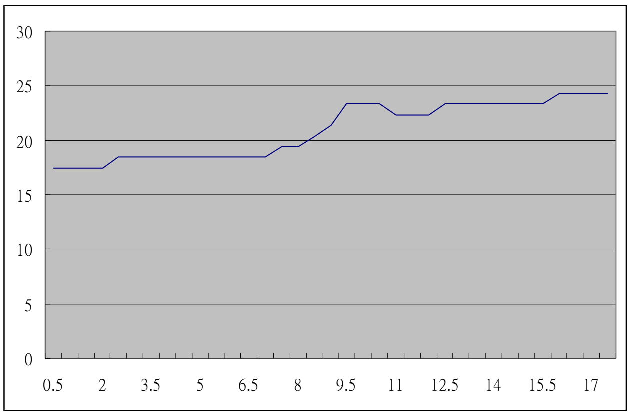
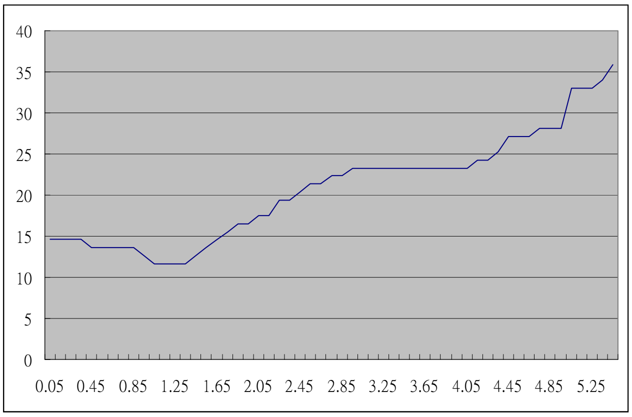


Table 4-4: Collision as a result of size exaggeration

(position scale: 2)

scale	# of sample	# of collision	%
0.05 ~ 0.35	103	15	14.56
0.45 ~ 0.85	103	14	13.59
0.95	103	13	12.62
1.05 ~ 1.35	103	12	11.65
1.45	103	13	12.62
1.55	103	14	13.59
1.65	103	15	14.56
1.75	103	16	15.53
1.85 ~ 1.95	103	17	16.5
2.05 ~ 2.15	103	18	17.48
2.25 ~ 2.35	103	20	19.42
2.45	103	21	20.39
2.55 ~ 2.65	103	22	21.36
2.75 ~ 2.85	103	23	22.33
2.95 ~ 4.05	103	24	23.3
4.15 ~ 4.25	103	25	24.27
4.35	103	26	25.24
4.45 ~ 4.65	103	28	27.18
4.75 ~ 4.95	103	29	28.16
5.05 ~ 5.25	103	34	33.01
5.35	103	35	33.98
5.45	103	37	35.92



4.3 Image metamorphosis

Caricature generation is formulated as a warping process in our approach. There are two different methods to warp an image. The first, called forward mapping, scans through the source image pixel by pixel, and copies them to the appropriate place in the destination image. The second, reverse mapping, goes through the destination image pixel by pixel, and samples the corresponding pixel from the source image. The most important feature of reverse mapping is that every pixel in the destination image gets set to something appropriate. In forward mapping, some pixels in the destination might not get painted, and will have to be interpolated. A feature-based image metamorphosis method [22], which calculates the image deformation as a reverse mapping, is employed in our system. The problem can be stated as: "Which pixel coordinate in the source image do we sample for each pixel in the destination image?" A pair of corresponding lines (one defined relative to the source image, the other defined relative to the destination image) in the source and destination images defines a coordinate mapping from the destination image pixel coordinate X to the source image pixel coordinate X' such that for a line PQ in the destination image and P'Q' in the source image.

$$u = \frac{(X-P)\cdot(Q-P)}{\|Q-P\|^2} \tag{19}$$

$$v = \frac{(X-P)\cdot Perpendicular(Q-P)}{\|Q-P\|} \tag{20}$$

$$X' = P' + u\cdot(Q'-P') + \frac{v\cdot Perpendicular(Q'-P')}{\|Q'-P'\|} \tag{21}$$

The value u is the position along the line, and v is the perpendicular distance in pixels from the line.

At this stage, we have two sources for metamorphosis: an artist's work and the disparity map. We will also maintain three facial meshes. One is the pre-annotated facial mesh of an artist's work. Another one is the subject's facial feature mesh extracted by AAM. The remaining one is the exaggerated facial feature mesh. We regard the corresponding line pairs of each mesh as references for the multiple-line algorithm (Fig.4-6). We then generate caricature images with exaggerated contour as well as the given drawing style and the disparity map. It is straightforward to change the drawing style simply by replacing the source images.

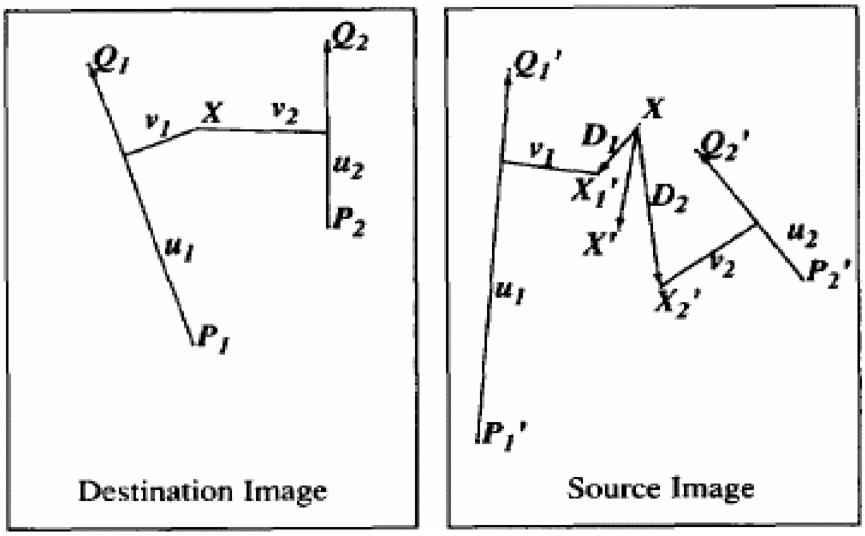
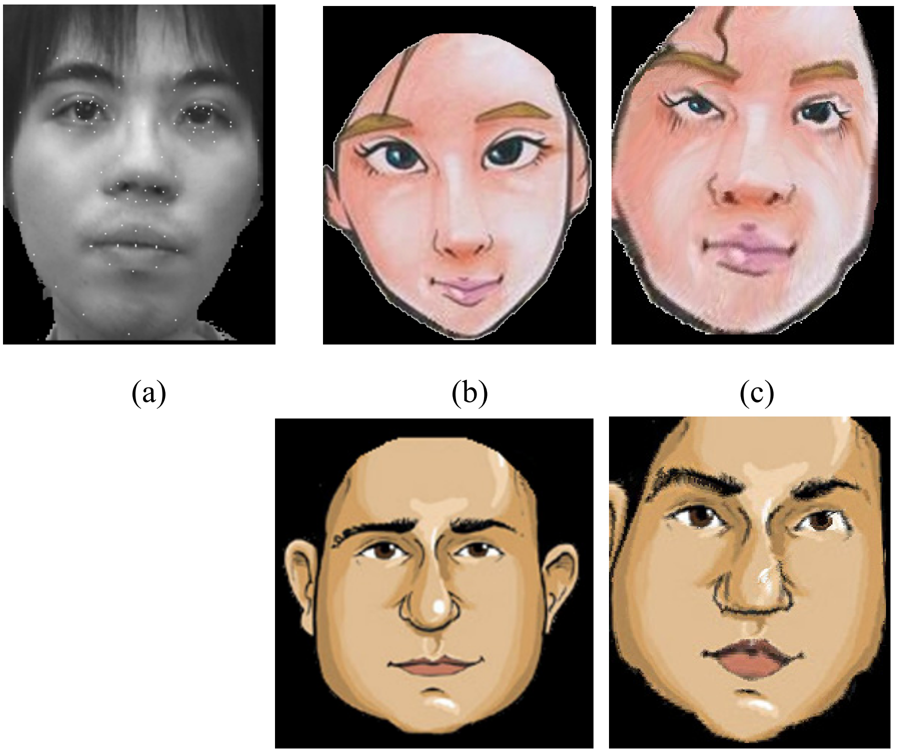


Fig. 4-6: Affected by all the line pairs, pixel X in the destination image is mapped to X' in the source image.



(a) (b) (c)

(d) (e)

Fig. 4-7: (a) subject (b,d) different artists' work (c,e) caricatures with (b,d)'s style.