A Hough Transform for Detecting Facial Symmetry Axis

Zheng, Qi
Faculty of Design, Kyushu University

Inoue, Kohei
Faculty of Design, Kyushu University

Ono, Naoki
Faculty of Design, Kyushu University

Hara, Kenji
Faculty of Design, Kyushu University

出版情報：Journal of the Institute of Industrial Applications Engineers. 10 (3), pp.53-59, 2022-07-25. Institute of Industrial Applications Engineers
バージョン：

Journal of the Institute of Industrial Applications Engineers Vol.10, No.3, pp.53–59, (2022.7.25)
DOI: 10.12792/JIIAE.10.53 Online edition: ISSN 2187-8811 Print edition: ISSN 2188-1758

Paper

A Hough Transform for Detecting Facial Symmetry Axis

Qi Zheng* Non-member, Kohei Inoue†* Member
Naoki Ono* Member, Kenji Hara* Member

(Received March 24, 2022, revised July 15, 2022)

Abstract: We propose a method for detecting a facial symmetry axis from a frontal face image. The proposed method adopts a voting procedure like Hough transform that transforms a set of edge points into perpendicular bisectors of two different edge points selected from the set. The initial guess of a symmetry axis is adjusted accurately by the following fine-tuning procedure. Experimental results show that the proposed method correctly detects facial symmetry axes from real face images visually and quantitatively.

Keywords: Face image, Symmetry axis, Hough transform, Perpendicular bisector, Edge detection.

1. Introduction

Human faces send out various messages such as their personalities and feelings to their surroundings. The current prevalence of online communication makes us recognize again the richness of the information brought by face-to-face communication [1]. For stable and reliable communication between human beings and/or machines, it is required to capture, recognize and present faces accurately.

One of the salient geometric features of faces is bilateral symmetry, with which a face or head can be divided into roughly mirror-image left and right halves by a midsagittal plane [2], and such a facial symmetry is believed to be a hallmark of appealing faces [3]. Therefore, detecting facial symmetry axis has a wide range of applications for face image processing including face recognition and generation. Until now, a large number of studies on symmetry detection have been conducted by many researchers. Bartalucci et al. [4] surveyed the methods for recognizing 2D/3D symmetry by the symmetry line and bilateral symmetry by the symmetry plane, and especially focused on 2D symmetry recognition of the human back and 3D symmetry recognition of the human face [4]. Park et al. evaluated the performance of state-of-the-art discrete symmetry detection algorithms, and indicated that symmetry detection in real-world images remains a challenging, unsolved problem in computer vision [5]. Xiao and Wu analyzed gradient-based and phase-based symmetry detectors, and suggested that the latter is preferable to human visual characteristics [6]. Sandhan and Choi incorporated a symmetry prior to their objective function for eyeglass reflection removal [7].

In this paper, we propose a method for detecting facial symmetry axis from a frontal face image. The proposed method can be viewed as a Hough transform [8] that consists of a transformation from an image plane into another parameter space, a voting and mode-seeking procedure, and the inverse transformation from the parameter space into the original image plane. Experimental results demonstrate that the proposed method detects facial symmetry axes from frontal face images faithfully, and reduces the standard deviation of a set of individual face images on average, which can be viewed as supporting evidence of successful facial alignment based on the symmetry axis.

The rest of this paper is organized as follows: Section 2 describes the proposed method for detecting facial symmetry axis from a frontal face image. Section 3 shows experimental results with real face images. Finally, Section 4 concludes this paper.

2. Proposed Method

In this section, we propose a method for detecting a symmetry axis from a frontal face image. The proposed method consists of two stages: the initial guess and fine-tuning stages. In the following subsections, we explain the details of those stages in order.

2.1 Initial Guess Let F be an image with a frontal face. Then the purpose of this work is to detect an axis of symmetry on the face in F. Let $G = [g_{ij}]$ be the grayscale image of F, where g_{ij} denotes the grayscale value of the pixel (i, j) in G for $i = 0, 1, \ldots, m-1$ and $j = 0, 1, \ldots, n-1$ where m and n denote the numbers of rows and columns in G, respectively. We first detect the edges in G to have a binary image $B = [b_{ij}]$, where $b_{ij} = 1$ indicates the edge pixel, and $b_{ij} = 0$ otherwise. The method we used for detecting edges is summarized in Appendix.

The basic idea of our symmetry axis detection is that if two points are line symmetry, the axis of symmetry is given by the perpendicular bisector of the line segment bounded by the two points. Let $\boldsymbol{p}_a = [i_a, j_a]^T$ and $\boldsymbol{p}_b = [i_b, j_b]^T$ are two different points or pixels in B with $b_{i_a,j_a} = b_{i_b,j_b} = 1$, where the superscript T denotes the matrix transpose. Then, as shown in Fig. 1, any point $\tilde{\boldsymbol{p}} = [\tilde{i}, \tilde{j}]^T$ on the perpendicular

† Corresponding: k-inoue@design.kyushu-u.ac.jp
* Faculty of Design, Kyushu University
4-9-1 Shiobaru, Minami-ku, Fukuoka 815-8540, Japan

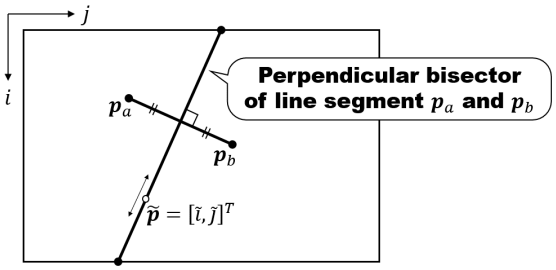


Figure 1: White point $\tilde{\boldsymbol{p}}$ denotes any point on the perpendicular bisector of line segment $\boldsymbol{p}_a$ and $\boldsymbol{p}_b$. If $\boldsymbol{p}_a$ and $\boldsymbol{p}_b$ are line symmetry, then the perpendicular bisector coincides with the axis of symmetry.

bisector of line segment $\boldsymbol{p}_a\boldsymbol{p}_b$ satisfies

$$\left(\frac{\boldsymbol{p}_b - \boldsymbol{p}_a}{\|\boldsymbol{p}_b - \boldsymbol{p}_a\|}\right)^T (\tilde{\boldsymbol{p}} - \boldsymbol{p}_a) = \frac{1}{2}\|\boldsymbol{p}_b - \boldsymbol{p}_a\|. \tag{1}$$

Multiplying the both sides of (1) by $2\|\boldsymbol{p}_b - \boldsymbol{p}_a\|$, we have

$$2\,(\boldsymbol{p}_b - \boldsymbol{p}_a)^T\,(\tilde{\boldsymbol{p}} - \boldsymbol{p}_a) = \|\boldsymbol{p}_b - \boldsymbol{p}_a\|^2, \tag{2}$$

or equivalently,

$$\begin{aligned} 2\,(\boldsymbol{p}_b - \boldsymbol{p}_a)^T\,\tilde{\boldsymbol{p}} &= \|\boldsymbol{p}_b - \boldsymbol{p}_a\|^2 + 2\,(\boldsymbol{p}_b - \boldsymbol{p}_a)^T\,\boldsymbol{p}_a \\ &= (\boldsymbol{p}_b - \boldsymbol{p}_a)^T\,(\boldsymbol{p}_b - \boldsymbol{p}_a + 2\boldsymbol{p}_a) \\ &= (\boldsymbol{p}_b - \boldsymbol{p}_a)^T\,(\boldsymbol{p}_b + \boldsymbol{p}_a)\,, \end{aligned} \tag{3}$$

which can be written with the elements of vectors as

$$2\left[(i_b - i_a)\,\tilde{i} + (j_b - j_a)\,\tilde{j}\right] = \begin{bmatrix} i_b - i_a \\ j_b - j_a \end{bmatrix}^T \begin{bmatrix} i_b + i_a \\ j_b + j_a \end{bmatrix}, \tag{4}$$

from which we have a relationship between $\tilde{i}$ and $\tilde{j}$ as

$$\tilde{j} = \frac{1}{j_b - j_a}\begin{bmatrix} i_b - i_a \\ j_b - j_a \end{bmatrix}^T \begin{bmatrix} \frac{i_b + i_a}{2} \\ \frac{j_b + j_a}{2} \end{bmatrix} - \frac{i_b - i_a}{j_b - j_a}\tilde{i} \tag{5}$$

under the condition that $j_b \neq j_a$, which is used for the voting procedure in Hough transform described below.

In Fig. 1, the perpendicular bisector intersects with the top and bottom boundaries of an image. We prepare two accumulators on the two boundaries, and vote to them the intersection points of the perpendicular bisector and the boundaries. Let $A^{\mathrm{top}} = [a_j^{\mathrm{top}}]$ and $A^{\mathrm{bot}} = [a_j^{\mathrm{bot}}]$ be the top and bottom accumulators, respectively, where a_j^{top} and a_j^{bot} are their elements initialized to 0. For two points $\boldsymbol{p}_a$ and $\boldsymbol{p}_b$, the horizontal axis $\tilde{j}^{\mathrm{top}}$ of the top intersection point is given by substituting $\tilde{i} = 0$ into (5). On the other hand, the horizontal axis $\tilde{j}^{\mathrm{bot}}$ of the bottom intersection point is given by substituting $\tilde{i} = m - 1$ into (5). Then we vote $(0, \tilde{j}^{\mathrm{top}})$ and $(m-1, \tilde{j}^{\mathrm{bot}})$ to A^{top} and A^{bot}, respectively, as

$$a_j^{\mathrm{top}} \leftarrow a_j^{\mathrm{top}} + \exp\left(-\frac{\left(j - \tilde{j}^{\mathrm{top}}\right)^2}{2\sigma^2}\right) \tag{6}$$

$$a_j^{\mathrm{bot}} \leftarrow a_j^{\mathrm{bot}} + \exp\left(-\frac{\left(j - \tilde{j}^{\mathrm{bot}}\right)^2}{2\sigma^2}\right) \tag{7}$$

for $j = 0, 1, \ldots, n-1$, where σ is a positive parameter for controlling the influence of $\tilde{j}^{\mathrm{top}}$ and $\tilde{j}^{\mathrm{bot}}$ to the neighboring elements of the accumulators. For all combinations of the pairs of edge pixels, we repeat this voting procedure. After that, we find the highest values in both accumulators as

$$j_*^{\mathrm{top}} = \arg\max_j \left(a_j^{\mathrm{top}}\right) \tag{8}$$

$$j_*^{\mathrm{bot}} = \arg\max_j \left(a_j^{\mathrm{bot}}\right), \tag{9}$$

and have a line segment bounded by $(0, j_*^{\mathrm{top}})$ and $(m-1, j_*^{\mathrm{bot}})$, which is an initial guess for the axis of symmetry.

2.2 Fine-Tuning Next, we improve the accuracy of the symmetry axis detection by using the result of the above initial guess as follows. Once we obtain the estimated axis of symmetry, we can map a point onto the other side of the axis. If there is another point near the mapped point, then we see that the corresponding points are nearly symmetrical in terms of the axis with a degree of confidence. On the basis of this observation, we propose a method for fine-tuning the axis of symmetry by attaching an importance to each edge pixel.

Let $\boldsymbol{t} = [0, \tilde{j}_*^{\mathrm{top}}]^T$ and $\boldsymbol{b} = [m-1, \tilde{j}_*^{\mathrm{bot}}]^T$ be the top and bottom intersection points of the initially estimated axis and image boundary. Then a point $\boldsymbol{p}_a$ is mapped to the opposite side of the symmetry axis by

$$\bar{\boldsymbol{p}}_a = \left[2\left(\frac{\boldsymbol{b} - \boldsymbol{t}}{\|\boldsymbol{b} - \boldsymbol{t}\|}\right)\left(\frac{\boldsymbol{b} - \boldsymbol{t}}{\|\boldsymbol{b} - \boldsymbol{t}\|}\right)^T - I\right](\boldsymbol{p}_a - \boldsymbol{t}) + \boldsymbol{t}, \tag{10}$$

where I denotes a 3×3 identity matrix. For another point $\boldsymbol{p}_b$, we define an importance value as follows:

$$v_{ab} = \exp\left(-\frac{\|\boldsymbol{p}_b - \bar{\boldsymbol{p}}_a\|^2}{2\sigma_s^2}\right), \tag{11}$$

where σ_s denotes a positive parameter for controlling the influence of $\bar{\boldsymbol{p}}_a$ to the neighboring points. With the importance value v_{ab}, we recompute the accumulators A^{top} and A^{bot}, the elements of which are initialized to 0s again, as

$$a_j^{\mathrm{top}} \leftarrow a_j^{\mathrm{top}} + v_{ab}\exp\left(-\frac{\left(j - \tilde{j}^{\mathrm{top}}\right)^2}{2\sigma^2}\right) \tag{12}$$

$$a_j^{\mathrm{bot}} \leftarrow a_j^{\mathrm{bot}} + v_{ab}\exp\left(-\frac{\left(j - \tilde{j}^{\mathrm{bot}}\right)^2}{2\sigma^2}\right), \tag{13}$$

instead of (6) and (7), respectively, where note that $\tilde{j}^{\mathrm{top}}$ and $\tilde{j}^{\mathrm{bot}}$ are the functions of $\boldsymbol{p}_a$ and $\boldsymbol{p}_b$ as described in (5). For a distant point $\boldsymbol{p}_b$ from $\bar{\boldsymbol{p}}_a$, which means that the point $\boldsymbol{p}_b$ is not symmetrical to $\boldsymbol{p}_a$, the value v_{ab} becomes small, and the distant point $\boldsymbol{p}_b$ has little influence on the computation of the accumulators. As a result, it is expected that we have more reliable accumulators than the initial ones. From the recomputed accumulators, we have the updated endpoints j_*^{top} and j_*^{bot} by (8) and (9). Such a fine-tuning procedure is repeated until both j_*^{top} and j_*^{bot} converge. The proposed procedure for detecting symmetry axis is summarized in Algorithm 1.

Algorithm 1 Symmetry Axis Detection

Require: frontal face image F
Ensure: horizontal axes j_*^{top} and j_*^{bot}
1: Make the grayscale image $G = [g_{ij}]$ of F for $i = 0, 1, \ldots, m-1$ and $j = 0, 1, \ldots, n-1$.
2: Detect edges in G to have a binary edge image $B = [b_{ij}]$.
3: Let $\boldsymbol{p}_k = [i_k, j_k]^T$ be the kth edge pixel in B with $b_{i_k,j_k} = 1$ for $k = 1, 2, \ldots, N$, where N denotes the number of edge pixels.
4: Initialize accumulators A^{top} and A^{bot} to zeros.
5: **for** $k = 1, 2, \ldots, N-1$ **do**
6: $\quad \boldsymbol{p}_a = \boldsymbol{p}_k$.
7: $\quad$ **for** $l = k+1, k+2, \ldots, N$ **do**
8: $\quad\quad \boldsymbol{p}_b = \boldsymbol{p}_l$.
9: $\quad\quad$ Compute $\tilde{j}^{\text{top}}$ by (5) with $\tilde{i} = 0$.
10: $\quad\quad$ Increment all a_j^{top} by (6).
11: $\quad\quad$ Compute $\tilde{j}^{\text{bot}}$ by (5) with $\tilde{i} = m-1$.
12: $\quad\quad$ Increment all a_j^{bot} by (7).
13: $\quad$ **end for**
14: **end for**
15: Compute j_*^{top} and j_*^{bot} by (8) and (9). ▷ Initial Guess
16: **while** True **do**
17: $\quad$ Set $j_{\text{old}}^{\text{top}} = j_*^{\text{top}}$ and $j_{\text{old}}^{\text{bot}} = j_*^{\text{bot}}$.
18: $\quad$ Initialize accumulators A^{top} and A^{bot} to zeros.
19: $\quad$ **for** $k = 1, 2, \ldots, N-1$ **do**
20: $\quad\quad \boldsymbol{p}_a = \boldsymbol{p}_k$.
21: $\quad\quad$ **for** $l = k+1, k+2, \ldots, N$ **do**
22: $\quad\quad\quad \boldsymbol{p}_b = \boldsymbol{p}_l$.
23: $\quad\quad\quad$ Compute v_{ab} by (11).
24: $\quad\quad\quad$ Compute $\tilde{j}^{\text{top}}$ by (5) with $\tilde{i} = 0$.
25: $\quad\quad\quad$ Increment all a_j^{top} by (12).
26: $\quad\quad\quad$ Compute $\tilde{j}^{\text{bot}}$ by (5) with $\tilde{i} = m-1$.
27: $\quad\quad\quad$ Increment all a_j^{bot} by (13).
28: $\quad\quad$ **end for**
29: $\quad$ **end for**
30: $\quad$ Compute j_*^{top} and j_*^{bot} by (8) and (9). ▷ Fine-Tuning
31: $\quad$ **if** $j_{\text{old}}^{\text{top}} = j_*^{\text{top}}$ and $j_{\text{old}}^{\text{bot}} = j_*^{\text{bot}}$ **then**
32: $\quad\quad$ Break out of the while loop.
33: $\quad$ **end if**
34: **end while**
35: **return** j_*^{top} and j_*^{bot}.

In this algorithm, the initial guess at line 15 is followed by iterative fine-tuning at line 30. The main difference between them is whether the importance value v_{ab} in (11) is used in the increment of accumulators (lines 25 and 27) or not (lines 10 and 12).

3. Experimental Results

In this section, we show experimental results of facial symmetry axis detection from real face images. Figure 2 shows an input face image, where Fig. 2(a) shows a color face image with 400 × 640 pixels (Wallpaper-House.com `https://wallpaper-house.com/wallpaper-id-307187.php`), and Fig. 2(b) shows the edge image extracted from the input image by the method described in Appendix.

Figure 3 shows the result of initial guess of the facial symmetry line, where the accumulators for the top and bottom boundaries are shown in Figs. 3(a) and (b), respectively, of which the vertical and horizontal axes denote the number

(a) Input frontal face image F

(b) Edge image B

Figure 2: Example of frontal face image: (a) Input color image F and (b) Binary edge image B obtained by the method described in Appendix, where the edge pixels are denoted by black pixels, and the other pixels are shown in gray.

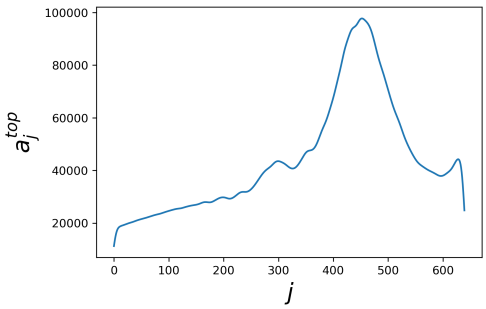
of votes a_j^{top} or a_j^{bot} and the horizontal location j. The parameter σ in (6) and (7) is set to 5. We extract the highest peaks from Figs. 3(a) and (b) by (8) and (9), and draw a red line segment from $(0, j_*^{\text{top}})$ to $(m-1, j_*^{\text{bot}})$ as shown in Fig. 3(c), which denotes the initial guess of the symmetry axis.

Next, Fig. 4 shows the result of fine-tuning procedure, where Figs. 4(a) and (b) show the accumulators for the top and bottom boundaries at the first iteration of the fine-tuning stage with $\sigma_s = 15$ in (11). In comparison with Figs. 3(a) and (b), Figs 4(a) and (b) are unimodal which provides a reliable estimation of mode. Figures 4(c) and (d) show detected symmetry axes at the first and last (13th) iterations, respectively. We can see that the red line moved to the position of facial symmetry axis correctly by fine-tuning. Figure 5 shows the inclination-corrected face after the fine-tuning.

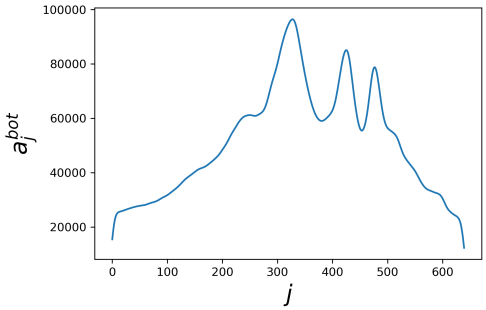
We also applied our method to the Japanese Female Facial Expression Database (JAFFE) [10], which consists of 213 images of different facial expressions from 10 different Japanese female subjects. Figure 6 shows six examples of facial symmetry axis detection (left) and inclination correction (right). We evaluate the effect of inclination correction by using standard deviation defined by

$$SD = \sqrt{\frac{1}{mn\Xi} \sum_{\xi=1}^{\Xi} \left\| G^{\xi} - \bar{G} \right\|_F^2}, \tag{14}$$

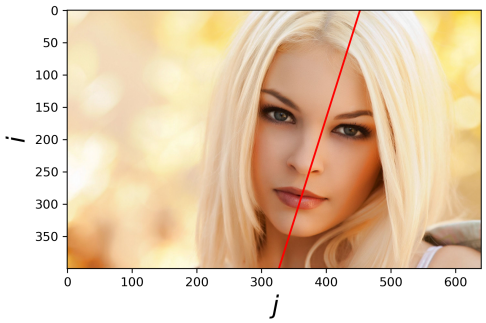
where G^{ξ} for $\xi = 1, 2, \ldots, \Xi$ denote Ξ face images of a person, and $\bar{G} = \frac{1}{\Xi} \sum_{\xi=1}^{\Xi} G^{\xi}$ denotes the mean image, and $\| \cdot \|_F$



(a) a_j^{top} vs j

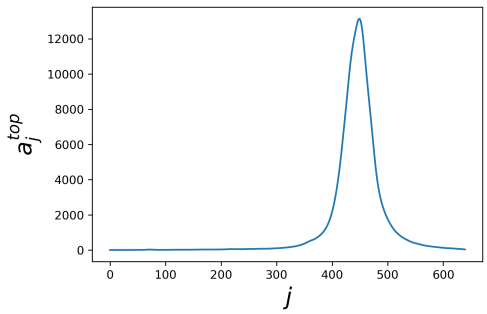


(b) a_j^{bot} vs j

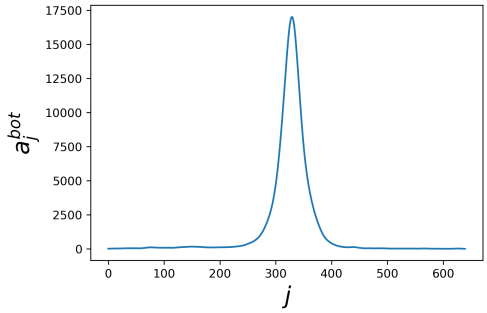


(c) Detected symmetry axis

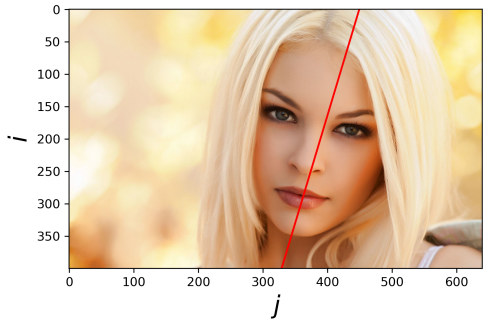
Figure 3: Initial guess: (a) Accumulator A^{top} for top endpoint, (b) Accumulator A^{bot} for bottom endpoint, and (c) the detected symmetry axis illustrated by red line.



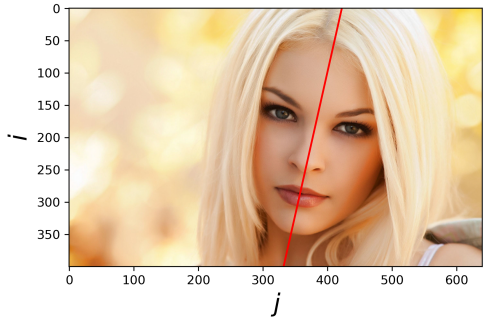
(a) Accumulator A^{top} at the 1st iteration



(b) Accumulator A^{bot} at the 1st iteration



(c) Detected symmetry axis at the 1st iteration



(d) Detected symmetry axis at the 13th iteration

Figure 4: Fine-tuning: (a) Accumulator A^{top} for top endpoint in the first iteration, (b) Accumulator A^{bot} for bottom endpoint in the first iteration, (c) the detected symmetry axis in the first iteration, (d) the final result after 13 iterations.

denotes the Frobenius norm [11]. It is expected that if faces are aligned faithfully then the value of SD becomes small. Figure 7 shows the standard deviation of a set of face images of each person, where the vertical and horizontal axes denote the standard deviation and the person ID numbers for 10 persons.

We cropped the central part of each image from 256×256 to 190×190 pixels to remove the void areas appeared by rotating the face image. The blue and orange bars denote the original and rotated (inclination-corrected) faces. We can see that the standard deviation decreased for 7 persons out of 10 by the proposed method. The values of the standard deviation are summarized in Table 1 by averaging them in the 10 persons, where smaller values are bolded. The proposed method reduced the standard deviation on average by correcting the inclination of faces.

Finally, we investigate the effect of shadows caused by asymmetric lighting. Figure 8 shows the results with the Yale Face Database B [12], where five cropped images of subject number 32 with pose number 00 are selected for azimuth 00, 10, 25, 50, and 70 with elevation 00, and laid

Figure 5: Inclination-corrected face.

Table 1: Comparison of (averaged) standard deviation.

	Original	Rotated
Person 1	**22.7**	22.8
Person 2	15.0	**14.8**
Person 3	22.9	**22.8**
Person 4	17.7	17.7
Person 5	21.6	**21.5**
Person 6	18.1	18.1
Person 7	21.6	**21.2**
Person 8	21.6	**21.0**
Person 9	15.9	**15.5**
Person 10	22.8	**22.7**
Averaged standard deviation	20.0	**19.8**

vertically. For the details of this dataset, please visit the website [13]. In Fig. 8, the left and right columns show the detected symmetry axes and inclination-corrected images, respectively. This figure shows that dark shadow can disturb the results of symmetry axis detection as observed in (e), (g), and (i) corresponding to azimuth +25, +50 and +70. For those shadowed face images, shadow removal methods such as the portrait shadow manipulation [14] and the unsupervised portrait shadow removal [15] can be used for the preprocessing of symmetry axis detection.

4. Conclusion

In this paper, we proposed a method for detecting the symmetry axis from a frontal face image by a voting procedure based on a Hough transform between an image space of binary images and a parameter space of perpendicular bisectors. Experimental results with real face images demonstrated the effectiveness of the proposed method visually and quantitatively by evaluating the standard deviation of obtained images.

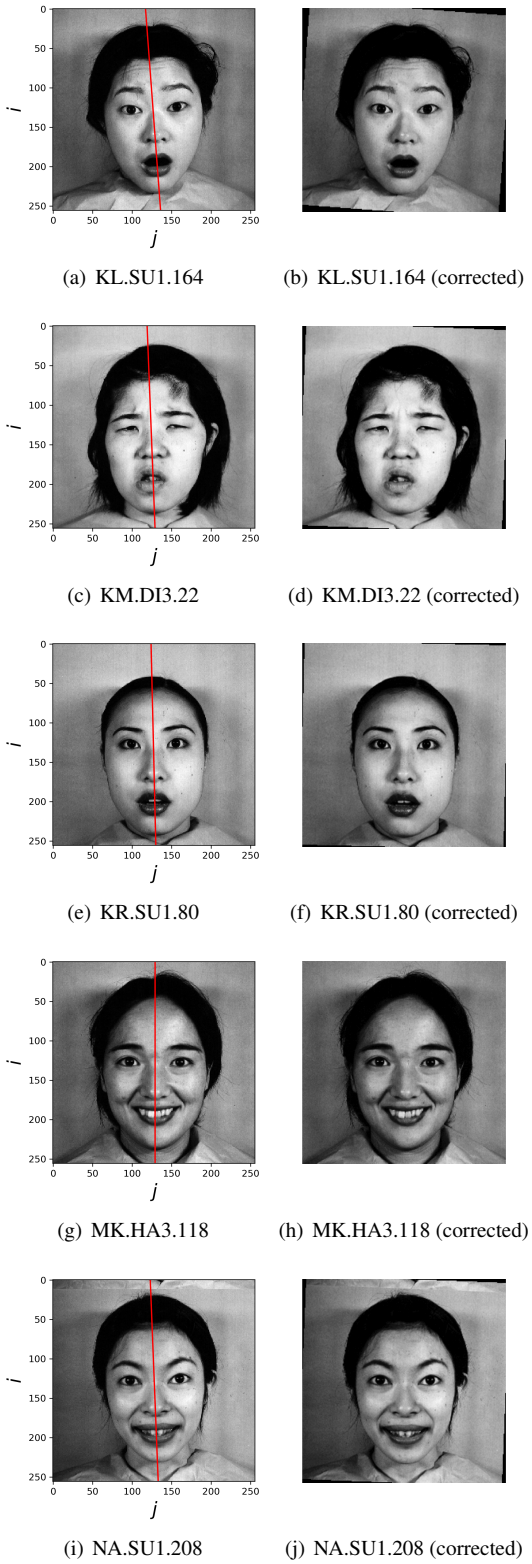
Our future work will include the application of the proposed method to facial beauty based on facial symmetry according to the results that facial attractiveness increased with the increase in facial symmetry by Rhodes et al. [16].

Appendix

From a grayscale image $G = [g_{ij}]$, we detect edges as follows: We first compute the horizontal and vertical gradients by the Sobel filter [9]:

$$dh_{ij} = g_{i-1,j+1} + 2g_{i,j+1} + g_{i+1,j+1} - g_{i-1,j-1} - 2g_{i,j-1} - g_{i+1,j-1} \quad (15)$$

$$dv_{ij} = g_{i+1,j-1} + 2g_{i+1,j} + g_{i+1,j+1} - g_{i-1,j-1} - 2g_{i-1,j} - g_{i-1,j+1}, \quad (16)$$



(a) KL.SU1.164 (b) KL.SU1.164 (corrected)

(c) KM.DI3.22 (d) KM.DI3.22 (corrected)

(e) KR.SU1.80 (f) KR.SU1.80 (corrected)

(g) MK.HA3.118 (h) MK.HA3.118 (corrected)

(i) NA.SU1.208 (j) NA.SU1.208 (corrected)

Figure 6: Symmetry axis detection and face rotation results on JAFFE dataset: (Left) Detected symmetry axes, and (Right) Inclination-corrected faces. Each caption shows the image file name including the subject's initial and expression label.

from which, we have the gradient magnitude $M_{ij} = \sqrt{dh_{ij}^2 + dv_{ij}^2}$. If $M_{ij} > \theta$ for a threshold value $\theta > 0$, we regard the pixel (i, j) as a

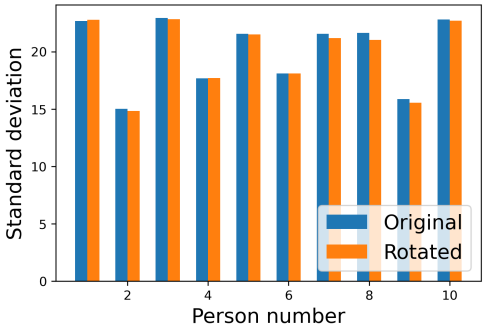


Figure 7: Standard deviation of individual image sets.

candidate of edge pixel, and compute the direction of the gradient $\phi_{ij} = \arctan(\frac{dv_{ij}}{dh_{ij}}) \in [-\pi, \pi]$ (rad), which is converted into $d_{ij} = 180\phi_{ij}/\pi \in [-180, 180]$ (deg). In the above experiments, we set $\theta = 100$ constantly. If $d_{ij} < 0$, then we add 180 to d_{ij} to map the interval $[-180, 0]$ to $[0, 180]$.

Finally, we perform non-maximum suppression to have a binary edge image $B = [b_{ij}]$ as follows: If $d_{ij} < 22.5°$ or $d_{ij} \geq 157.5°$, then we judge that the edge is horizontal. In this case, if $M_{ij} > M_{i-1,j}$ and $M_{ij} > M_{i+1,j}$, then we judge that the pixel (i, j) is an edge pixel and set $b_{ij} = 1$. If $22.5° \leq d_{ij} < 67.5°$, then we judge the edge is left diagonal. In this case, if $M_{ij} > M_{i-1,j-1}$ and $M_{ij} > M_{i+1,j+1}$, then we also set $b_{ij} = 1$. If $67.5° \leq d_{ij} < 112.5°$, then we judge that the edge is vertical. In this case, if $M_{ij} > M_{i,j-1}$ and $M_{ij} > M_{i,j+1}$, then we also set $b_{ij} = 1$. Finally, if $112.5° \leq d_{ij} < 157.5°$, then we judge the edge is right diagonal. In this case, if $M_{ij} > M_{i-1,j+1}$ and $M_{ij} > M_{i+1,j-1}$, then we also set $b_{ij} = 1$. The remaining pixels in B have the value 0, which means the non-edge pixel.

References

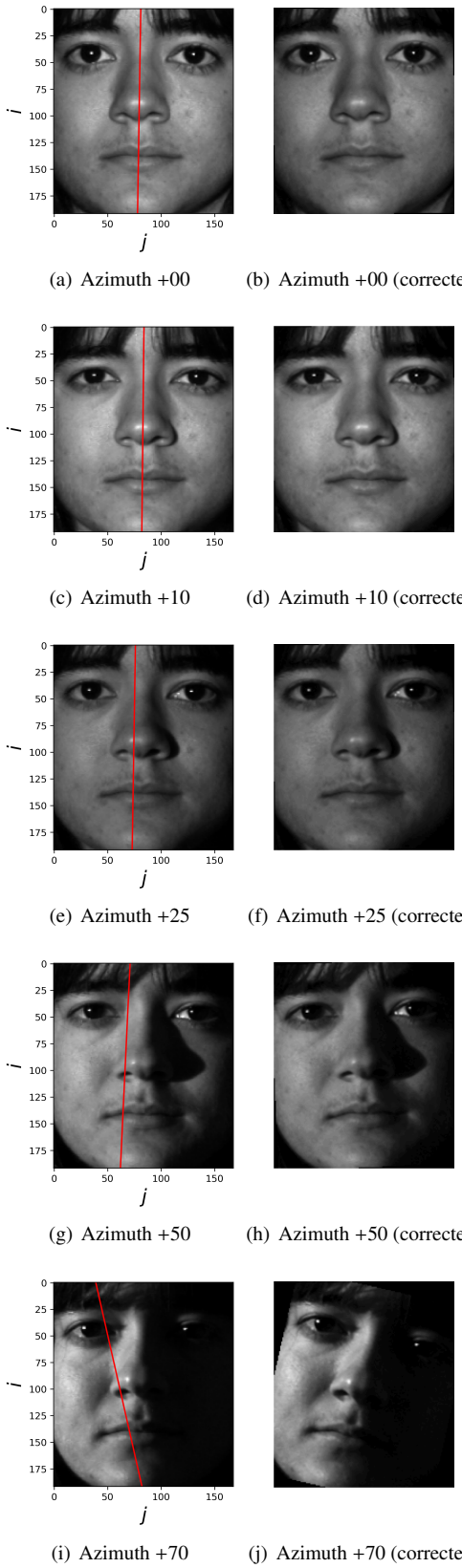
[1] R. Paul, J. Sharrard, S. Xiong, "The Importance of Face-to-Face Communication in the Digital World," *Journal of Nutrition Education and Behavior*, vol. 48, no. 10, p. 681, 2016. DOI:10.1016/j.jneb.2016.09.014.

[2] Biologydictionary.net Editors. "Midsagittal Plane." Biology Dictionary, Biologydictionary.net, `https://biologydictionary.net/midsagittal-plane/`, access date: 18 Jun. 2019.

[3] E. Prokopakis, I. Vlastos, V. Picavet, G. N. Trenite, R. Thomas, C. Cingi, and P. Hellings, "The golden ratio in facial symmetry," *Rhinology*, vol. 51, no. 1, pp. 18-21, 2013. DOI:10.4193/Rhino12.111

[4] C. Bartalucci, R. Furferi, L. Governi, Y. Volpe, "A Survey of Methods for Symmetry Detection on 3D High Point Density Models in Biomedicine", *Symmetry*, vol. 10, no. 7: 263, 2018. DOI:10.3390/sym10070263

[5] M. Park, S. Lee, P.-C. Chen, S. Kashyap, A. A. Butt, Y. Liu, "Performance evaluation of state-of-the-art discrete symmetry detection algorithms", *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2008. DOI: 10.1109/CVPR.2008.4587824



(a) Azimuth +00 (b) Azimuth +00 (corrected)

(c) Azimuth +10 (d) Azimuth +10 (corrected)

(e) Azimuth +25 (f) Azimuth +25 (corrected)

(g) Azimuth +50 (h) Azimuth +50 (corrected)

(i) Azimuth +70 (j) Azimuth +70 (corrected)

Figure 8: Effect of shadows caused by asymmetric lighting: (Left) Detected symmetry axes, and (Right) Inclination-corrected faces. Each caption shows the value of azimuth in degrees.

[6] Z. Xiao and J. Wu, "Analysis on Image Symmetry Detection Algorithms," *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)*, 2007, pp. 745-750, DOI: 10.1109/FSKD.2007.173.

[7] T. Sandhan and J. Y. Choi, "Anti-Glare: Tightly Constrained Optimization for Eyeglass Reflection Removal," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp.1675-1684, DOI: 10.1109/CVPR.2017.182.

[8] R. O. Duda, P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures", *Commun. ACM*, vol. 15, no. 1, pp.11-15, 1972. DOI: 10.1145/361237.361242

[9] I. Sobel, "An Isotropic 3x3 Image Gradient Operator", 2014. `https://www.researchgate.net/publication/239398674_An_Isotropic_3x3_Image_Gradient_Operator`

[10] M. Lyons, S. Akamatsu, M. Kamachi and J. Gyoba, "Coding facial expressions with Gabor wavelets," *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, pp. 200-205, DOI: 10.1109/AFGR.1998.670949.

[11] Matrix norm, *Wikipedia*, `https://en.wikipedia.org/wiki/Matrix_norm`, access date: 2022.3.17

[12] Kuang-Chih Lee, J. Ho and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 684-698, 2005. DOI: 10.1109/TPAMI.2005.92.

[13] The Yale Face Database B. `http://vision.ucsd.edu/~leekc/ExtYaleDatabase/Yale%20Face%20Database.htm`, access date: 2022.7.8

[14] X. Zhang, J. T. Barron, Y.-T. Tsai, R. Pandey, X. Zhang, R. Ng, D. E. Jacobs, "Portrait shadow manipulation", *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 78:1-78:14, 2020. DOI: 10.1145/3386569.3392390

[15] Y. He, Y. Xing, T. Zhang, Q. Chen, "Unsupervised Portrait Shadow Removal via Generative Priors", *MM '21: Proceedings of the 29th ACM International Conference on Multimedia*, pp. 236-244, 2021. DOI: 10.1145/3474085.3475663

[16] G. Rhodes, F. Proffitt, J. M. Grady, A. Sumich, "Facial symmetry and the perception of beauty", *Psychonomic Bulletin & Review*, vol. 5, pp. 659-669, 1998. DOI:10.3758/BF03208842

Qi Zheng (Non-member) She received B.E. degree from Southwest Jiaotong University in 2020. She is currently taking a pre-master's course at Kyushu University. Her research interest is image processing.

Kohei Inoue (Member) He received B.Des., M.Des. and D.Eng. degrees from Kyushu Institute of Design in 1996, 1998 and 2000, respectively. He is currently an Associate Professor at Kyushu University. His research interests include pattern recognition and image processing.

Naoki Ono (Member) was born in Japan, in 1961. He received a Ph.D. degree in engineers from Kyushu University in 1997, and is presently an associate professor at Kyushu University. He has worked on image processing and pattern recognition. He is a member of IIAE and IEICE.

Kenji Hara (Member) He received the B.E. and M.E. degrees from Kyoto University in 1987 and 1989, respectively, and the Ph.D. degree from Kyushu University in 1999. He is currently a Professor at Kyushu University. His research interests include physics-based vision and geometric modeling.