Robust Face Recognition for Multi-View Video

Dr.B.S.Rao[1], V.Sony[2], B.BalaVenu[3], M.D.S.Prasad[4], A.D.Bhujangarao[5]
Swarnandhra College of Engineering and Technology, Narsapur

Abstract- The frontal face detection development has matured for application to everyday life. However, practically deals with various profile face views are still difficult. To build different detector a common solution is to organized by a decision tree and clarified that each detector handles a single view or a few views. However, the training images collection for each face view is laborious and time consuming. Moreover, because of insufficient training images many profile face detectors cannot perform as well as the frontal face detector.

Many detectors management also leads to too complex a logic structure to classify a face into its corresponding view. In this research paper, we propose a novel method to reuse a frontal face detector to detect multi-view faces, which do not need any data collection or training processes for various profile views. We aim to extend its application to multi-view faces and focus on exploiting the potential of a general frontal face detector. We perform a theoretical analysis to explain why our methodology works from different perspectives, and implement the proposed method based on the sliding window strategy. Furthermore, by a genetic algorithm the searching process is optimized with an original fitness function. Experimental results verify that our proposed method can successfully detect human faces in almost all head poses in the dataset containing a complete collection of head poses in yaw and pitch axes.

Keywords- multi-view face detection; flipping scheme; frontal face detector.

I. INTRODUCTION

With the rapid advancement of computing power and the availability of modern sensors, analysis and response equipment, and technologies, computers are becoming increasingly capable and intelligent. Many research achievements and commercial applications have demonstrated the natural way for a computer to interact with a human by observing people through cameras,listening through microphones, discriminating these inputs, and reacting appropriately in a friendly manner.

Face detection is one of the fundamental techniques to enable this natural and intelligent human – computer interaction, which does not completely rely on traditional devices such as keyboards, mice, and displays. Computers need to find and understand the human face well before they can begin to comprehend human thoughts and intentions truly. The cornerstone of all applications revolving around automatic facial image analysis is face detection including, but not limited to, face recognition and verification [7], face tracking for surveillance [17], facial behaviour analysis [3], facial attribute recognition [16,18,26], face relighting and morphing [15], facial shape reconstruction [2], content-based image/video retrieval [24], and organization and presentation of digital photo albums [23].

The arbitrary image is given, the goal of face detection is to determine whether there are any faces in the image and, if present, return the image location and extent of each face. Although this is one of the visual tasks that humans can do effortlessly, it is a very challenging task for computers, and has been one of the top research topics in the past few decades. In realistic application scenarios, the difficulty associated with face detection Can be attributed to the variability in scale, location, orientation (in-plane rotation),and pose (out-of-plane rotation). Facial expression, occlusion, presence/absence of structural components (beards, mustaches, or glasses), and lighting conditions also change the overall appearance of faces. Unlike other existing face detection strategies, in this study, we originally and successfully transform a retrained frontal face detector (trained with positive samples and negative samples of the frontal face) into a profile face detector without any data collection or training processes.

The remaining of this paper is organized as follows. Section 2 introduces the recent advances in the face detection and face dataset and summarizes the contributions of this work. Section 3 explains how the proposed method is motivated from a common phenomenon and developed to practical use in multi-view face detection. Section 4 presents the implementation of the proposed method and a speeding-up module using a genetic algorithm (GA). Section 5 reports the experimental results on both dataset of discontinuous images and the continuous image sequence of a video.In Section 6 we see conclusion and future work.

II. BACKGROUND

a. Advances in face detectors: Hundreds ofapproaches to face detection have been reported. Early works before the year 2000 have been comprehensively summarized by Helms *et al.* [5] and Yang *et al.* [6]. For instance, the various methods in that period were grouped into four categories [6]: knowledge-based methods, feature-invariant approaches, template-matching methods, and appearance-based methods. Knowledge-based methods use predefined rules to locate faces based on human knowledge of what constitutes a typical face; feature-invariant approaches aim to find facial structure features that are robust to pose, viewpoint, and lighting variations; template-matching methods use restored face patterns to judge if an image is a face; appearance-based methods learn face models from a set of representative training face images to perform detection. Appearance-based methods seem to show superior performance compared to the other methods and dominate the recent advances in face

detection, which could be attributed to the rapidly growing computation power and data storage.

Follow-up works on face detection from the year 2000 to the year 2015 have been surveyed by Stefanos*et al.* [27]. The Viola – Jones face detection methodology has had the most impact on face detection algorithms in the 2000s. To build a successful face detector this methodology has three main ideas which can run in real time: the integral image, classifier learning with AdaBoost, and the attentional cascade structure. The methodology also motivates many of the recent advances whose techniques are categorized into two general schemes: rigid templates, learned mainly via boosting-based methods or by the application of deep neural networks, and deformable models that describe the face by its parts. The general practice of these methods is to collect a large set of face and non-face samples and adopt certain machine learning algorithms to learn a face model to perform classification. Hence, the amount and quality of the training data can highly influence the performance of a detector.

b. Advances in face datasets: Modern detectors caneasily detect near-frontal faces (faces with slight out-of-plane rota-tions) and are widely used in real-world applications, such as in digital camera and electronic photoalbums. However, some intricate factors, such as extreme pose and large portion of occlusion, can cause large visual variations in the facial appearance. This problem is getting increasing attention, and many efforts are made to collect and annotate a considerable number of images in unconstrained conditions. The scale of face detection datasets has also developed from a few hundred faces to several hundreds of thousand faces, with increasing variations in pose and occlusion.However, most of the publicly available datasets for assessing face detection performance do not divide the number of faces in different view-points equally. For instance, LFW [12], Pascal Faces dataset [19], FDDB [20], Annotated Faces in-the-Wild [22], and WIDER Face dataset [29] (as in Fig. 1) only claim that facial images were collected in the wild with a high degree of variability in scale, pose, and occlusion. Considering that the faces in these datasets were captured or collected in natural scenes, the proportion between the number of near frontal faces and the number of faces in extreme poses may varies considerably. Using these datasets for training, the face detectors will probably perform better on near-frontal faces than extreme poses. The limitations of the datasets have partially contributed to the failure of some algorithms while handling heavy occlusion and atypical poses.Moreover, several datasets collect an equal number of facial images for different poses, such as the Pointing'04 head pose

Fig.1: Sample images of the wider face dataset.

Face images in this dataset have a high degree of variability in scale, pose, occlusion, expression, mark-up, and illumination. However, it does not claim equal proportion between different poses, or other aspects of variability. The marked bounding boxes are the annotated ground truth

Fig.2: Sample images of a person from the Pointing'04 head pose database.

The numbers of face image for different poses are the same, which current face detection datasets cannot ensure. From left to right, the horizontal angle h takes values of -90° , -45° , 0° , $+45^\circ$, $+90^\circ$. From top to bottom, the vertical angle v takes values of $+60^\circ$, 0° , -60°image database [8] and the CMU Multi-PIE face database [14]. As shown in Fig. 2, the head poses of the Pointing'04 database are determined by two angles (h, v): the horizontal angle varies from -90° to 90° in 15° intervals, and the vertical angle varies from -90° to 90° in 30° intervals. To obtain different poses, markers were placed in the whole collection room. Each marker corresponds to a pose (h, v). The whole set of markers covers a half-sphere in front of the person. The person was asked to stare successively at the different markers by adjusting the chair without moving his/her eyes. However, the shortcoming of Pointing'04 is that all the images are taken in the same controlled condition of simple background, illumination, and scale.

The CMU Multi-PIE face database records subjects under 15 view points and 19 illumination conditions while displaying a range of facial expressions. This database uses a system of 15 $^\circ$cameras. Thirteen cameras are located at head height, spaced in 15 intervals. Two additional cameras are located above the subject, simulating a typical surveillance view. Although the CMU Multi-PIE face database covers several hundreds of subjects, different poses, illuminations, and expressions, similar face sizes and backgrounds make it unsuitable for face detection algorithms. Both databases are frequently used for face recognition algorithms or head pose estimation. To our knowledge, few reported algorithms use either database as a face detection benchmark or a training dataset. Therefore, we can draw the following inferences according to the above information:

- Frontal faces or near-frontal faces are probably more frequently captured and collected in natural scenes than extreme poses. The proportion between the number of near-frontal faces and the number of faces in other poses may vary considerably in publicly available datasets.

- It is difficult for publicly available datasets to cover all view-points in the 3D space.
- Modern face detectors are trained using a training dataset where different poses are unevenly distributed, which means that a different pose accounts for a different proportion.
- Modern face detectors probably perform better on frontal faces and near-frontal faces than on other poses, especially extreme poses, which is reasonable considering that their performance heavily relies on the training data including face samples and non-face samples.

***c. Contributions of this work*:** Considering that thedetection of near-frontal faces is more reliable than faces with large out-of-plane rotations and the dataset of near-frontal faces is also easier to collect and annotate, in this paper we propose a completely different scheme called the 'flipping scheme' todetect multi-view faces by reusing the currently mature frontal face detector. The main contributions of the flipping scheme can be summarized as follows:

- Unlike other multi-view face detection strategies, the pro-posed flipping scheme does not need to collect multi-view face images.
- By reusing the currently mature frontal face detector, the flipping scheme does not need a training process, which is a key process to most face detection algorithms.
- The flipping scheme can detect almost all faces with out-of-plane rotations (verified in Section 5).

With all these merits, the proposed scheme handles the problem of multi-view face detection from a new perspective, whereas many types of research focus on getting the training dataset larger and training process is more complicated. Given that frontal face detection is already developed to achieve a reliable accuracy and an applicable processing speed, reusing the frontal face detector to detect multi-view faces will provide a new solution to this research direction.
However, the frontal face detector is trained especially to detect frontal faces and the profile face detector to detect profile faces. It sounds unreasonable to detect profile faces using the frontal face detector. The next section explains how the proposed scheme is motivated from a common phenomenon and developed to practical use in multi-view face detection.

III. METHODS FOR MULTI-VIEW FACE DETECTION

Motivation from mirror reversal: The proposedscheme to detect multi-view faces using only a frontal face detector is motivated by the common phenomenon of mirror reversal. Mir-ror reversal usually refers to the recognized left-to-right reversal of a mirror image Takano [28] conducted psycho-optic analyses to show that various kinds of mirror reversal can be reasonably explained within a consistent theoretical framework of the multi-process theory. Suppose

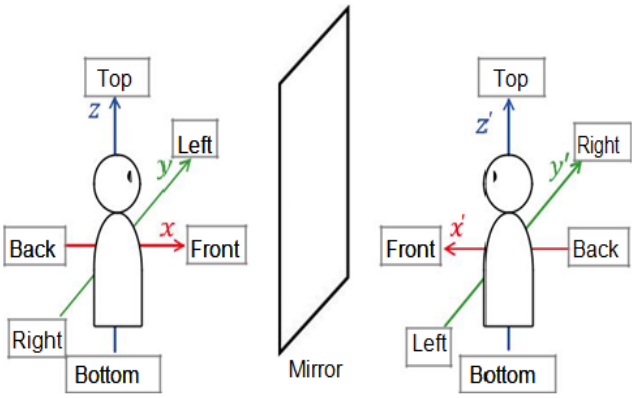
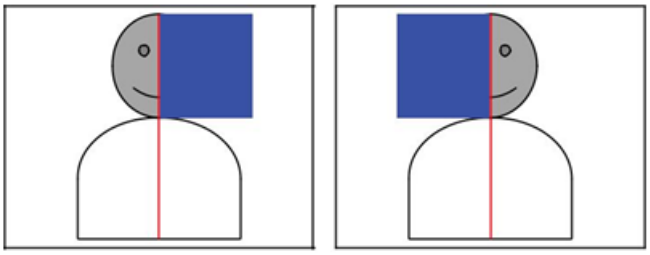


Fig.3: A type of left – right mirror reversal according to the multi-process theory.

The real viewer is drawn in actual line, and the mirror image in dotted line. The parallelogram represents a mirror. The texts describing the direction bounded in gray boxes (left, right, front, back, top, and bottom) regard the real viewer or the mirror image correspondingly as the view point



(a) (b)

Fig.4: Illustration of the difference between the picture of a real viewer in the frontal view and the picture of his/her mirror image.

(a) The picture of the real viewer. (b) The picture of the mirror image. The dotted line in red represents the approximate symmetryaxis. The face of the viewer is half occluded by the blue box is equivalent to a combination of two geometric transformations: translation of the viewer's coordinate system from the real viewer to the mirror image, and a 180° rotation around the z -axis. This rotation reverses both the x -axis and the y-axis, while leaving the z -axis intact.

Frontal view: When we look at ourselves througha mirror (suppose the gaze direction is orthogonal to the mirror surface), it is easy to mistake left for right. Takano and Tanaka
[11]found that 33% of the 102 viewers did not recognize their left – right mirror reversal when they faced the mirror because they did not conduct the mental process of viewpoint transformation. Similarly, when we look at a picture of an upright human face, it is difficult to recognize the left half of the face for the right half. This is because human faces are approximately symmetrical around the vertical axis in the middle, and the symmetry axis is parallel to the surface of the mirror/picture. The left half and right half of the face look

alike and contain very similar depth information. In the task of face detection, it is enough for a face detector to distinguish whether a candidate region is a human face or not. Therefore, either the left half or the right half of the human face can provide a face detector with all the necessary information of the whole face.

As shown in Fig. 4(a), the left half-face of the viewer is occluded and only the right half-face is visible. Here, the direction of left and right is based on the viewpoint of the viewer. Considering that the mirror image is left – right reversed, the right half-face is occluded in the picture of the mirror image as in Fig. 4(b). Neither Fig. 4(a) nor Fig. 4(b) can be detected by any frontal face detector. However, if we manually combine the left half-face of Fig. 4(a) and the right half-face of Fig. 4(b), it will result in a complete face that can be recognized as a human face. In this way, the particular problem to detect vertically half-occluded faces is solved by simply flipping the candidate region horizontally and recognizing whether the combined region is a frontal face or not.The phenomenon of mirror reversal provides evidence and explanation to why this strategy works:

- Human beings are conscious that faces are left – right sym-metric and are accustomed to observing left – right-reversed faces by mirrors/pictures. According to Ref. [11], some viewers even did not recognize that there is a left – right mirror reversal. The experiments in this literature also claim high similarity between the left and the right half of the face.
- In the problem of face detection, it is fine to assume that faces can be reconstructed by doubling either the left half or the right half.
- The reconstruction process does not affect the face detection performance of a human vision

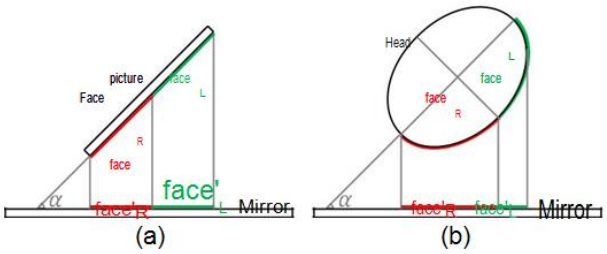


Fig.5: Illustration of projecting a face picture and a human head in profile view to a mirror.

- (a) The picture of a frontal-view face is projected onto a mirror. (b) A human head is projected onto a mirror. This figure illustrates both cases by a top view. The mirrors and the face picture are symbolized by thin rectangles. For simplicity, the human head is approximately symbolized by an eclipse. The green line represents the left half-face, denoted by faceL; the red line represents the right half-face, denoted by faceR. The corresponding projections on the mirror are denoted by faceLandfaceR, respectivelythe performance of a face detector trained with positive and negative samples. However, it may affect the result of face recognition and lead to a wrong identity, which will not be discussed in this paper.er does it affect

Profile view: Suppose that the angle between theleft – right plane of the viewer and the mirror surface, denoted byα, takes a value above $0^{\circ}$and below 90°. As shown in Fig. 5(a),the picture of a frontal-view face is projected onto a mirror. Both the left half-face $face_L$ and the right half-face $face_R$ are narrowed to the same extent, which makes their widths equivalent to each other. The projection to the original picture can be restored by resizing the width. The appearance does not change significantly after this simple geometrical transformation.

However, in Fig. 5(b), the human head is approximately sym-bolized by an ellipse for simplicity while observing from the top. $face_L$ and $face_R$ are represented by two symmetrical arcs. When the head is projected to the mirror with the inclined angle α, the width of $face_R$ becomes larger than that of $face_L$. $face_R$ contains the appearance information in 3D space that cannot be seen from a frontal-view picture. $face_L$ is also projected from the appearance information in 3D space, but it is more complicated: some region is invisible while observing from the orthogonal direction to the mirror surface; some region is occluded by convex parts, for instance, he nose; the other region is projected from an approx-imate spherical surface to a flat surface. When α is reduced to 0° , the widths of $face_L$ and $face_R$ become equivalent, which is the same as the mirror image of a frontal-view face picture as in

Fig. 5(a). In contrast, when α is increased to 90° , $face_R$ reaches its maximum and $face_L$ disappears.

The projection on the surface of the mirror can be regarded as an image capturing the face of a profile view. The term 'profile view' refers to the poses with various out-of-plane rotations, which

is symbolized by $\alpha \in [-90, 0) \cup (0, 90]$. As shown in Fig. 6, a frontal face ($\alpha = 0^{\circ}$) and two profile faces ($\alpha = 45^{\circ}$ and $\alpha = 90^{\circ}$) are reconstructed by flipping their projections of a half-face (asintroduced in Section 3.1.1). To reconstruct the faces in these cases for detection, one of $face_L$ and $face_R$ is better than the other while $|\alpha|$ is larger than 0°. However, asαincreases from $0^{\circ}$to 90°, theappearance of the reconstructed faces (Fig. 6(b), (c), (e), (f), and

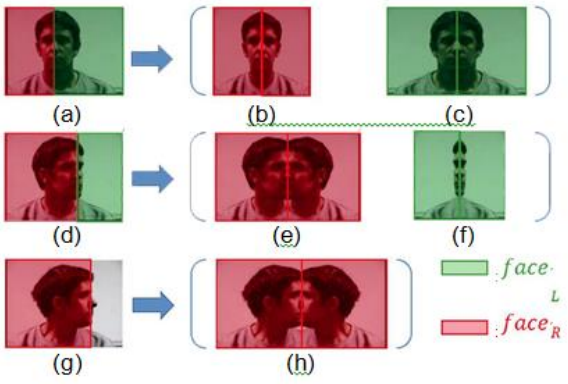


Fig.6: Illustration of reconstructing a whole face with either half of the face region. (a) The image of a frontal-view face ($\alpha = 0^{\circ}$).

(d) The image of a profile face (α= 45°). (g) The image of a profile face (α= 90°). (b), (e), and (h) Reconstruction by flipping the right half-face of sub-figure (a), (d), and (g), respectively. (c) and (f) Reconstruction by flipping the left half-face of sub-figure (a) and (d), respectively. The translucent green rectangle represents the region that contains $face_L$; the translucent red rectangle represents
the region containing $face_R$

Potential ability of training Starting with theViola – Jones face detection methodology, there have been many advances to obtain rigid templates by learning via boosting-based methods or by the application of deep neural networks. These works contain two general procedures: preparation of the datasets and learning the face detector. The performance of the detector depends on the diversity of the dataset and the adopted machine-learning algorithm. In the dataset for learning, there are positive samples, namely all kinds of faces that we aim to find, and negative samples, namely all kinds of non-faces that we want to exclude. Facial regions are cropped out or marked out to make the positive samples. Other parts of the upper body, such as hair, ear, neck and arms, are not included, because they change easily and frequently for different people, different occasions, or different poses. Only facial parts, such as eyes, nose, and mouth, retain the same pattern of relational locations, scales, and similar appearances

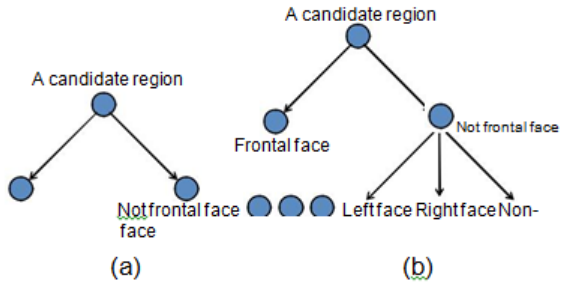


Fig.7: Illustration of classifying a candidate region into several classes.
(a) The case of general frontal face detection.
(b) The case of multi-view face detection using the flipping scheme

Suppose that there is an arbitrary image and no prior information is available. The general frontal face detection just classifies the candidate regions into two classes as in Fig. 7(a), namely 'frontal face' and 'not-frontal face'. In the flipping scheme, each candidate region within the image is supposed to be classified into one of the four classes as in Fig. 7(b): 'frontal face', 'left face' (the profile view whose left half-face can be detected), 'right face' (the pro-file view whose right half-face can be detected), and 'non-face'. The two additional classes have to be generated by repeating the frontal face detection process on the reconstructed faces.
The reconstructed faces can be divided into two cases according to the flipping direction. In one case, the candidate region is assumed as a 'left face', flipped to right, and combined with the original candidate region to construct a among different situations. In contrast, negative samples are supposed to cover all cases in which the detector may make mistakes. This explains why the newly collected datasets for face detection grow larger and claim adding more scenes in the past few decades. In the problem of detecting reconstructed faces first proposed in this paper, there is no dataset that demonstrates adding the reconstructed faces into the negative samples or positive samples. However, the appearance of these reconstructed faces (Fig. 6(b), (c), (e), and (h)) around the symmetry axis looks similar to a real frontal face. Even the most difficult case of Fig. 6(h) shares similar pixel-varying patterns around the eyes, nose, and mouth. Moreover, the facial skin accounts for the majority of the face region in both real faces and reconstructed faces. When the head turns from the frontal view to the profile view, the facial skin changes slightly in appearance. Therefore, there is little difference in the facial skin between the real faces and the reconstructed faces. The following section implements the proposed scheme and uses experiments to prove that the profile faces can be detected by the frontal face detector.

Flipping scheme algorithm The face detectionproblem is solved by performing some small classification problems on all candidate regions of an image to distinguish faces from non-faces. Faces are summarized and reported to the final output.
conjectured full face. If the conjectured full face is classified as 'frontal face', it proves that the assumption is correct, and the candidate region should be classified as 'left face' in the final output. In the other case, the candidate region is assumed as a 'right face', flipped to the left, which also makes a conjectured full face. If the conjectured full face in this case is classified as 'frontal face', it implies that the candidate region should be classified as 'right face' in the final output.
While flipping a candidate region to make these two cases of conjectured full face, flipping twice to make two separate conjectured full faces and detecting twice (as in Fig. 8(a) – (c)) is time consuming. We propose to reduce the process of the second level of Fig. 7(b) to flipping once and detecting once (Fig. 8(d) and (e)). As shown in Fig. 8, the candidate region is flipped to left and right separately to generate two conjectured full faces, while in the proposed idea it is flipped to left and right at the same time to generate a single reconstructed region that may contain several conjectured full faces. Figure 8(e) only needs to be detected once to judge which class the candidate region belongs to. The reconstructed region is treated as a new temporary image, and the location of a detected face corresponds to a different one in the original image.
Given an arbitrary image I and a frontal face detector, the following process flow explains how to apply the flipping scheme to detect multi-view faces:

- For each candidate region C (x_I , y_I , w_I , h_I) represented bythe left-top vertex (x_I , y_I), the width w_I , and the height h_I , flip it to the left (denoted by $f_L[C]$) and right (denoted by$f_R[C]$);

- Make a temporary image containing all the reconstructedregions, which is symbolized as $I = f_L[C] \cup C \cup f_R[C]]$;
- Perform frontal face detection with the frontal face detectorand obtain its returned detection result. For each returned frontal face region $F(x_I, y_I, w_I, h_I)$, follow the steps below to restore its coordinates and size of I to correspond-ing with those in I;

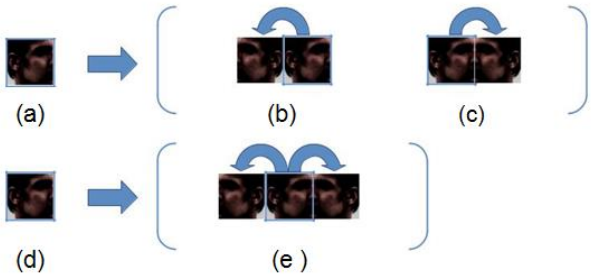


Fig.8:

- If it satisfies $F \subseteq C$, return a frontal face that can be specifiedby the rectangle of $(x_I+x_I-w_I, y_I+y_I, w_I, h_I)$.

- If it satisfies $F\ f_R(C)$, $F\ C \cup f_R(C)$, and $F \subseteq (f_L(C) \cup C)$, return a left face that can be specified by the rectangleof $(x_I+x_I+w_I-w_I, y_I+y_I, x_I+w_I-w_I, h_I)$.

•If it satisfies $F \quad f_L(C)$, $F \quad f_L(C) \cup C$, and $F \subseteq (C \cup f_R(C))$, return a right face that can be specified by therectangle of $(x_I+x_I-w_I, y_I+y_I, 2 \times w_I-x_I, h_I)$.

- If there are no faces detected on I by the frontal face detector, classify the current candidate region C as 'non-face' and skip to the next candidate region.

IV. IMPLEMENTATION OF MULTI-VIEW FACE DETECTION

a. Frontal face detector: This paper uses the SURF Cascade [21] for frontal face detection. It is derived from the Viola – Jones framework but adopts a different type of image feature instead of the Haar-like features. It applies the speeded-up robust features (SURF) [13] as a local scale- and rotation-invariant descriptor for object detection and does not use the key point detector part. It was reported to be smaller in model-size because of less number of cascade stages than the Viola – Jones detection framework and comparable to the state-of-the-art frontal face detection algorithm on both accuracy and processing speed [25]. Then, the detector is obtained by training a large-scale dataset with the boosting algorithm Gentle AdaBoost, which produces the best results among all variants according to [4].

b. Optimized searching with genetic algorithm: There are many candidate regions in a general object detection problem considering that the targets may be located at any possible positions, scales, and rotation angles. In this paper, the sliding window strategy is applied to search and check all possibilities, which results in time-consuming calculations. The face detection problem can be regarded as a searching problem for the parameters of the face, such as the location, size, and rotation angle. This paper applies the genetic algorithm to optimize the searching process of the targets to obtain a higher processing speed at the cost of sacrificing some accuracy.

GA is one of the evolutionary computation methods simulating the evolution of life to obtain an optimized solution [9]. The population consists of individuals. Each individual carries a solution coded in the chromosome. Each chromosome consists of genes. Each gene defines a search domain for a parameter. GA can search several search domains at the same time. It is necessary to calculate the fitness value to the environment for each solution obtained from the search domains. The individual with a low fitness value will be eliminated, while individuals with high fitness value will be selected probabilistically to create the new offspring and survive to the next generation. In the generation iteration, new individuals need to be created by the mutation rate to maintain the biological diversity. This is one of the vital operations to prevent convergence at a local optimum. Unlike the biological evolution, we adopt the elite saving strategy to save an elite individual and inherit it to the next generation. Therefore, the fitness value of the elite individual will not decrease even if the generation is iterated.

The simple GA is always combined with template matching for detection and does not require pre learning with the database with massive images during the applications in the field of computer vision [10]. However, template matching is restricted to detecting targets whose intensity distributions are similar to those of the template. Moreover, the learned cascade classifier is a strong classifier that consists of many weak classifiers distinguishing between positive samples and negative samples. As a result, the cascade classifier has learned the best separating capacity from the database. Furthermore, the weak classifiers are well organized (cascaded) so that the classification order is optimized to achieve the fastest classification. This paper first combines GA with the flipping scheme to detect multi-view faces. In our implementation, each chromosome consists of five genes coding the parameters to specify a candidate region, including the centre point, two scale factors for the x - and y-axes, and the in-plane-rotation angle. GA is utilized to optimize the process of searching for the optimal parameter combination.

c. ***Fitness value***: The fitness value is normally calculated from the fitness function, which is designed to evaluate the generated solutions. Some individuals with a large fitness value are selected to create a new offspring and survive to the next generation. In the case of face detection, it is necessary to out-put a directly comparable fitness value. In the SURF cascade framework, logistic regression is chosen as the weak classifier. It performs as a linear classifier that outputs the probability score (in the range 0 – 1) at any stage. We define the fitness value of an arbitrary candidate region C as $\text{fitness}(C) = s(C) + p(C)$, where $s(C)$ is the number of

passed stages and $p(C)$ is the probability output at the exit stage. Figure 9 displays the fitness value distribution of the proposed method on a sample image where there is no frontal face reported by the SURF cascade framework.

d. ***Evolutionary video processing***: The genes of the First generation are initialized randomly at the same time with the Fig. 9. Fitness value distribution of the proposed method on a sample image where there is no frontal face reported by the SURF cascade detection framework. (a) The original image. (b) Fitness values of the candidate regions on the image. The value on each pixel represents the fitness value of the candidate region centred at the pixel, with size (60, 60) and rotation angle 0°

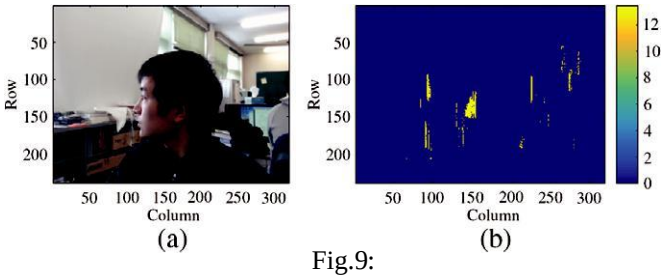


Fig.9:

evolutionary parameters such as the number of individuals, the number of generations, the crossover rate, and the mutation rate. In the case of detecting faces in several discontinuous images, both the genes of the first generation and the evolutionary parameters have to be re-initialized to process a different image. However, the target may only slightly change in continuous video frames. The genes of the first generation for the current frame can be inherited from the last generation of the previous frame. The candidate region containing a detected target in the previous frame will probably become an elite region, or generate an elite offspring within the first several generations. The newly appeared targets in the current frame will be detected in the same way as the detection process in the discontinuous images. The inherited information of the disappeared targets can be regarded as a random initialization for the current frame.

e. ***GA-optimized flipping scheme***: When the flippingscheme is combined with GA, the detection is treated as an optimization problem solved with the evolutionary method. The processing speed will be faster than the traditional sliding window strategy theoretically. The process flows of the GA-optimized flipping scheme are given below:

- Generate individuals of the population of the first generation randomly. In evolutionary video processing, generate the first generation randomly for the first frame, and inherit from the previous frame for the current frame.
- For each individual, decode the chromosome into parameters to specify a candidate region in the image. To deal with the candidate region, adjust the detection size range of SURF cascade according to the scale factor of this individual. Then, follow the steps introduced in Section 3.3 for the candidate region. Count the number of overlapped positive rectangles and assign it to the fitness value.

- Perform genetic operators such as reproduction, crossover, and mutation to generate the next generation.
- Check the terminating criterion and output the results of the elite individuals.

V. EXPERIMENTAL RESULTS

***a. Experiment on discontinuous images*:** Theexperiments introduced in this section were designed to verify whether the proposed method can detect multi-view faces or not, and how much the inclined angle α can be increased by detecting multi-view faces successfully. We implemented the flipping scheme with the sliding window strategy to search and check all possibilities. We also implemented the flipping scheme optimized by GA to obtain faster processing at the cost of sacrificing some accuracy. The SURF cascade detection framework with its default parameters (the number of nearest neighbours: k= 3 for KNN; minimum target size: 32 × 32; scale rate: 120%) is used as the frontal face detector. The Pointing'04 dataset (15 people, 93 poses, 1395 images) is selected for the experiments, because it consists of a complete collection of all human head poses in the yaw and pitch axes by small intervals (15° for yaw; 30° for pitch). The number of images for each pose and each test subject is equivalently balanced, which is perfect for testing the influence of head poses to a face detector. Furthermore, the images for each person are not organized continuously. We implemented the proposed method by regarding them as individually discontinuous images. To evaluate the results, the accuracy is calculated by counting the proportion of successfully detected faces including the left faces, the right faces, or the frontal faces. All results are tested with Visual Studio 2010 (C++) on a PC with Intel Core i5 – 3570 CPU (3.40 GHz) and 8 GB memory. The ranges of the MULTI-VIEW FACE DETECTION regions are as follows:

- The range of position: any possible coordinates that make the candidate region a valid part of the images, horizontally from 0 to 320 (image width), and vertically from 0 0to 240 (image height). The unit of image size used in this paper is the pixel.

- The range of size: from 32 × 32 (the minimum size that SURF cascade detector can handle) to 160 × 160 (the size set manually that is larger than the maximum face size in the dataset). The flipping scheme with the sliding window method uses the scale ratio of 110% between two neighboring sizes. The flipping scheme with GA searches the scale continuously.

- The range of in-plane rotation angle: the flipping scheme with sliding window method rotates the image from 0° to 360° in intervals of 10° . The flipping scheme with GA searches in a continuous range of [0° , 360°).

As shown in Figs. 10 and 11, the accuracies of all three detection methods are calculated for each yaw angle or pitch angle using the images of all people and the corresponding head poses. The frontal face detector of the SURF cascade only detects a small range at around 0° for both yaw and pitch. It detects frontal faces (pitch: 0° ; yaw: 0°) with maximum accuracy. The accuracy decreases heavily when the absolute value of the yaw or pitch angle grows. The proposed flipping scheme with the sliding window strategy achieves the best performance while the flipping scheme optimized with GA (crossover rate: 90%; mutation rate: 5%; the number of generation: 100; population: 100) decreases a little. The flipping scheme with the sliding window strategy reaches an average processing speed of 9.52 fps, and the flipping scheme optimized with GA reaches 59.70 fps on the same dataset. These results verify that the proposed methods can detect human faces in almost all head poses collected in the Pointing'04 dataset. Hence, the proposed method changed the frontal face detector into a multi-view face detector successfully without additional image collection and training.

b. Video processing experiments: The proposedmethod optimized with GA is also tested on an image sequence

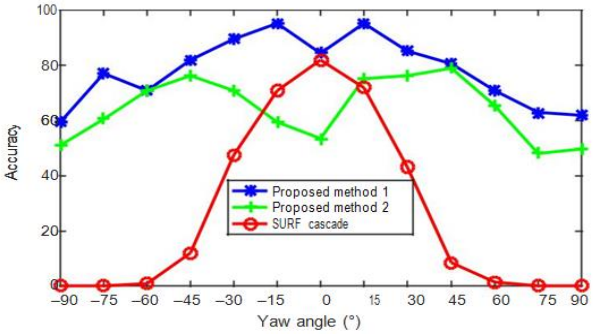


Fig.10: Performance comparison between the frontal face detector and our proposed methods over the yaw angle.

VI. PROPOSED METHOD

1: flipping scheme with the sliding window strategy; proposed method 2: flipping scheme optimized by the GA. In the horizontal axis, each value represents a yaw angle. Each yaw angle cor-responds to a series of head poses sharing the same yaw angle in the dataset. Each head pose corresponds to the images of different people.

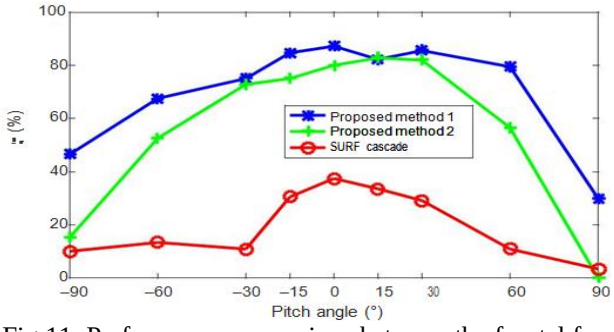


Fig.11: Performance comparison between the frontal face detector and our proposed methods over the pitch angle. Proposed method

1: flipping scheme with the sliding window strategy; proposed method 2: flipping scheme optimized by the GA. In the horizontal axis, each value represents a pitch angle. Each pitch angle corresponds to a series of head poses sharing the same pitch angle in the dataset. Each head pose corresponds to the imagesof different people.

Table I. Results of video processing experiments using the flipping scheme optimized by GA (crossover rate: 90%; mutation rate: 5%)

Number of Generation	Population	Accuracy (%)	Processing time (ms)
100	100	92.98	2078.04
50	50	78.30	565.48
10	100	63.83	263.35
20	20	60.43	100.08
10	10	56.38	25.87

with a total number of images of 470 and the image size of 320 × 240 collected in the context of a laboratory. The video processing mode is activated, which initialized the genes of the first generation for the current frame by inheriting from the last generation of the previous frame. Theoretically, the performance of GA will be equivalent to that of the sliding window strategy if the number of generations and the population (number of individuals per generation) are large enough. However, these two parameters can be set small to obtain a higher speed at the cost of a small decrease in accuracy. As shown in Table I, the decrease in the number of generations and population leads to a decrease in accuracy and a higher speed.

The proposed method optimized with GA uses the genetic information (five genes) to represent the parameters of a candidate region: the center point, two scale factors (for the horizontal and vertical axis, respectively), and a rotation angle. As the evolution proceeds, individuals with the highest fitness values obtain the highest probability to pass copies of their genes on to successive generations. When the generation number is large enough, the individual with the highest fitness value in the last generation carries the fittest genetic information, which represents the best combination of the five parameters. In this way, the rotation problem could be easily solved by GA, while it is more complex and takes more time on computation by using the sliding window strategy. It implies that the flipping scheme optimized by the GA is more suitable for tasks that require higher speed and do not care about a small decrease in accuracy. Figure 12 shows some examples of the detection results using the implementation optimized by GA. These successfully detected examples show that the proposed method can handle the extreme head poses. Moreover, this image sequence is not captured in a controlled condition of a simple background or illumination, which proves the adaptability of the proposed method

Fig.12: Some examples of successfully detected results using the proposed method optimized by the GA

VII. CONCLUSION

In this paper, we proposed a novel algorithm to use the flipping scheme to transform the frontal face detector into a profile face detector for various profile views. The proposed method reuses the frontal face detectors and does not need data collection or training processes for the other face views. We also provided a theoretical explanation from different perspectives on why the flipping scheme works and can change the frontal face detector into a multi-view face detector. Then, we implemented the flipping scheme with the sliding window strategy and verified that the proposed method could detect human faces in almost all head poses collected in the Pointing'04 dataset. We also implemented the flipping scheme optimized by the GA, which is more suitable for tasks that require higher speed and do not care about a small decrease in accuracy. In the future, we plan to conduct experiments to test the flipping scheme on the multi-target detection problem. Moreover, the flip-ping scheme proposed in this paper did not take advantage of any particular information on the human face, and it has potential appli-cation to the detection problem of any other symmetric objects.

VIII. ACKNOWLEDGMENTS

The authors would like to thank Prof. S. Kaneko for providing helpful suggestions about mirror reversal. This work was supported in part by JSPS KAKENHI Grant Number JP16K01647. Facial photos of Figs. 1, 2, 6, and 8 are from the publicly available database of WIDER [18] and pointing'04

Figures 9 and 12 are from a video sequence dataset which captured photos of the first author in our own lab. Publishing consent was obtained from the subject himself for identifying the facial photos.

IX. REFERENCES

[1]. Takano Y. Why does a mirror image look left-right reversed? A hypothesis of multiple processes. *Psychonomic Bulletin & Review* 1998; **5**:37 – 55.

[2]. Blanz V, Vetter T. A morphable model for the synthesis of 3d faces. *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999; 187 – 194.

[3]. Pantic M, Rothkrantz L. Automatic analysis of facial expressions: the state of the art. *IEEE Transactions on Pattern Analysis and MachineIntelligence* 2000;**22**(12):1424 – 1445.

[4]. Friedman J, Hastie T, Tibshirani R. Additive logistic regression: A statistical view of boosting. *Annals of Statistics* 2000; **28**:337 – 407.

[5]. Hjelmas E, Low B. Face detection: a survey. *Computer Vision andImage Understanding* 2001;**83**:236 – 274.

[6]. Yang M, Kriegman D, Ahuja N. Detecting faces in images: a survey. *IEEE Transaction on PAMI* 2002;**24**(1):34 – 58.

[7]. Zhao W, Chellappa R, Phillips J, Rosenfeld A. Face recognition: a literature survey. *ACM Computing Surveys* 2003; **35**(4):399 – 458.

[8]. Gourier N, Crowley J. Estimating face orientation from robust detection of salient facial structures. *Proceedings of Pointing 2004*, ICPR, International Workshop on Visual Observation of Deictic Gestures, 2004.

[9]. Stephen K, Fukumi M, Akashi T, Akamatsu N. Optimizing feature extraction for the camera mouse using genetic algorithms. *WorldScientific and Engineering Academy and Society Transaction on Computers* 2006;**11**(5):2722 – 2726.

[10]. Akashi T, Wakasa Y, Tanaka K, Stephen K, Fukumi M. Genetic eye detection using artificial template. *Journal of Signal Processing* 2006; **10**(6):453 – 463.

[11]. Takano Y, Tanaka A. Mirror reversal: empirical tests of compet-ing accounts. *Quarterly Journal of Experimental Psychology* 2007; **60**:1555 – 1584.

[12]. Huang GB, Ramesh R, Berg T, Learned-Miller E. Labeled faces in the wild: a database for studying face recognition in unconstrained environments. University of Massachusetts, Amherst, Technical Report, 07 – 49, 2007.

[13]. Herbert B, Andreas E, Tinne T, Luc V. Speeded-up robust fea-tures (SURF). *Computer Vision and Image Understanding* 2008; **110**(3):346 – 359.

[14]. Gross R, Matthews I, Cohn J, Kanade T, Baker S. Multi-PIE. *Imageand Vision Computing* 2009;**28**(5):807 – 813.

[15]. Wang Y, Zhang L, Liu Z, Hua G, Wen Z, Zhang Z, Samaras D. Face relighting from a single image under arbitrary unknown lighting conditions. *IEEE Transactions on Pattern Analysis and MachineIntelligence* 2009;**31**(11):1968 – 1984.

[16]. Kumar N, Berg A, Belhumeur P, Nayar S. Attribute and simile classifiers for face verification. *IEEE 12th International Conferenceon Computer Vision*, 2009; 365 – 372.

[17]. Kalal Z, Mikolajczyk K, Matas J. Face-TLD: tracking-learning-detection applied to faces. *2010 IEEE International Conference onImage Processing*, 2010; 3789 – 3792.

[18]. Fu Y, Guo G, Huang T. Age synthesis and estimation via faces: a survey. *IEEE Transactions on Pattern Analysis and MachineIntelligence* 2010;**32**(11):1955 – 1976.

[19]. Everingham M, Gool LV, Williams CK, Winn J, Zisserman A. The pascal visual object classes (voc) challenge. *International Journal ofComputer Vision* 2010;**88**(2):303 – 338.

[20]. Jain V, Learned-Miller E. FDDB: a benchmark for face detec-tion in unconstrained settings, Technical Report, University of Mas-sachusetts, Amherst, 2010.

[21]. Li J, Wang T, Zhang Y. Face detection using SURF cascade. *2011 IEEE International Conference on Computer Vision Workshops*,pp.2183 – 2190, 2011.

[22]. Koestinger M, Wohlhart P, Roth PM, Bischof H. Annotated facial landmarks in the wild: a large-scale, real-world database for facial landmark localization, First IEEE International

[23]. Workshop on Bench-marking Facial Image Analysis Technologies, 2011.
[24]. Kemelmacher-Shlizerman I, Shechtman E, Garg R, Seitz S. Exploring photo bios. *ACM Transactions on Graphics (TOG)* 2011; **30**(4):61.
[25]. Wu Z, Ke Q, Sun J, Shum H. Scalable face image retrieval with identity-based quantization and multi reference re-ranking. *IEEETransactions on Pattern Analysis and Machine Intelligence* 2011;**10**:1991 – 2001.
[26]. Li J, Zhang Y. Learning SURF Cascade for fast and accurate object detection. *Proceedings of the 2013 IEEE Computer SocietyConference on Computer Vision and Pattern Recognition*, 2013;3468 – 3475.
[27]. Laurentini A, Bottino A. Computer analysis of face beauty: a survey. *Computer Vision and Image Understanding* 2014;**125**:184 – 199.
[28]. Stefanos Z, Zhang C, Zhang Z. A survey on face detection in the wild: past, present and future. *Computer Vision and Image Understanding* 2015; **138**:1 – 24.
[29]. Takano Y. Mirror reversal of slanted objects: a psycho-optic expla-nation. *Philosophical Psychology* 2015; **28**(2):240 – 259.
[30]. Yang S, Luo P, Loy CC, Tang X. WIDER FACE: a face detec-tion benchmark. *IEEE Conference on Computer Vision and PatternRecognition (CVPR)*, 2016.