

*Research Paper*

## Fast Method for Face Detection Based on the Characteristic of Cascade Classifier

TOMOAKI YOSHINAGA,<sup>†1</sup> SHIGEKI NAGAYA  
and ISAO KARUBE<sup>†1</sup>

We propose a fast method of face detection that is based on the characteristics of a cascade classifier that is called a cascade step search (CSS). The proposed method has two features. First is a gradual classification that uses only a few layers of the cascade classifier to estimate the face-likelihood distribution. Second is an efficient search that uses the face likelihood distribution. The search is operated at intervals that are optimally changed according to the likelihood. This reduces the number of sub-windows that must be processed. The face-likelihood on the image window exposes the likelihood near the window and the next scaled one. These features can reduce the costs of classifications in the face detection process. Our experiments on face detection show that the proposed method is about five times faster than the traditional searches and maintains a high detection rate.

### 1. Introduction

Face detection is a fundamental technology with a lot of applications. The detected faces are used in many image-processing applications, for example, face enhancement, face identification, facial-expression recognition, and gaze estimation. Face detection plays a major part in the pre-processing phase of these applications. Fast computation for face detection is therefore a must so that the applications dealing with face information can run on embedded systems.

One of the most popular fast methods of face detection was developed by Viola and Jones<sup>1)</sup>. One of the major contributions of their work is the use of a cascade classifier as a face detector. The cascade can significantly speed up the classification time by using a coarse-to-fine strategy, which is formed from an easy classification for obvious non-face images and a hard-classification strategy

for possible face images. Since their work, several other fast methods that use a cascade classifier have been proposed. Those methods proposed stronger classifiers composing a cascade. Their cascade can detect a face faster and more accurately than the traditional one. A strong classifier is created by using high-quality visual features<sup>2),3)</sup> or capable functions for the classification<sup>4)</sup>. On the other hand, Brendan<sup>7)</sup> optimized each layer of the cascade, thereby speeding up the detection process.

The above-described methods attempt to speed up the “face classification” for a sub-window clipped from an image. Generally, the face detection process consists of scanning all the sub-windows on an image, and face classification is executed on each sub-window. The scanning is defined as a “face search”, in contrast to a “face classification”. In most common face detection methods, the same face search process is executed. It is a very simple process of scanning all the image positions and scales at particular intervals. In the present work, we aim at speeding up the face detection process, so we focus on the face search method using a cascade classifier.

In the general object detection process, the search method is strongly related to the classification process. For example, when using the template matching algorithm, Murase<sup>10)</sup> stated that he could reduce the operational cost by using the max limit of possibility gained from the classification process. Accordingly, we investigated the characteristics of a cascade classifier and devised a fast search method that uses these characteristics. We devised a fast-face-search method called the “cascade step search (CSS)” that is based on two characteristics of a cascade classifier, namely, the number of cascade layers outputting a “true” equals face-likelihood and the distribution of the likelihood in terms of the position and scale. Using these characteristics, CSS can reduce the number of sub-windows scanned on an image while maintaining the face detection rate.

### 2. Characteristic of Cascade Classifier

#### 2.1 Cascade Classifier

The cascade classifier created by AdaBoost training<sup>1)</sup> is used as the detector for the face detection process. The cascade consists of several classifier layers subordinately joined to each other (**Fig. 1**).

---

<sup>†1</sup> Central Research Laboratory, Hitachi, Ltd.

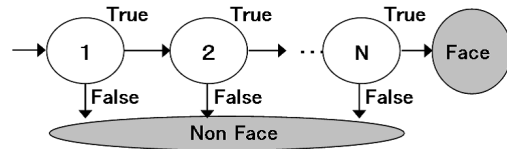


Fig. 1 Cascade classifier.

In the face classification by using a cascade, each layer of the cascade classifies the sub-window images as either positive (true) or negative (false). If the initial layer of the cascade outputs a true, the classification process is carried over to the second layer. The process continues to carry over as long as each layer outputs a “true”. Finally, the sub-window is judged as a face when the last layer outputs a “true”. By contrast, when a layer outputs a “false”, the process is instantaneously finished, and the sub-window is judged as a “non-face”. Accordingly, the non-face region can be quickly classified by using only a few classifiers. This contributes to the fast method of face detection because it is a rare-event detection problem; that is, non-face regions occupy almost the entire image.

The classification cost  $C_{cas}$  incurred when using a cascade classifier is given as

$$C_{cas} = C_1 + C_2 p_1 + C_3 p_1 p_2 + \dots \quad (1)$$

$$= C_1 + \sum_{i=2}^N \left( C_i \prod_{j<i} p_j \right), \quad (2)$$

where  $C_i (i = 1 \dots N)$  is the number of weak classifiers in the  $i$ -th layer of the cascade,  $N$  is the number of layers in the cascade, and  $p_i$  is the positive ratio of the  $i$ -th layers. Here,  $C_i$  increases when  $i$  increases. Generally,  $C_0$  is one order of magnitude, and  $p_i$  is about 50%; therefore, the majority of the sub-windows are classified as non-faces using only a few classifiers. On the other hand, the classification cost  $C_{noncas}$  incurred by a non-cascaded classifier is given as

$$C_{noncas} = \sum_{i=1}^N C_i. \quad (3)$$

The expression indicates that all the weak classifiers are used for all the sub-windows in an input image. It is thus concluded that a cascade classifier can significantly reduce the operating cost.

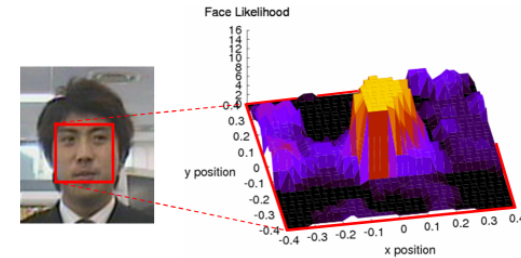


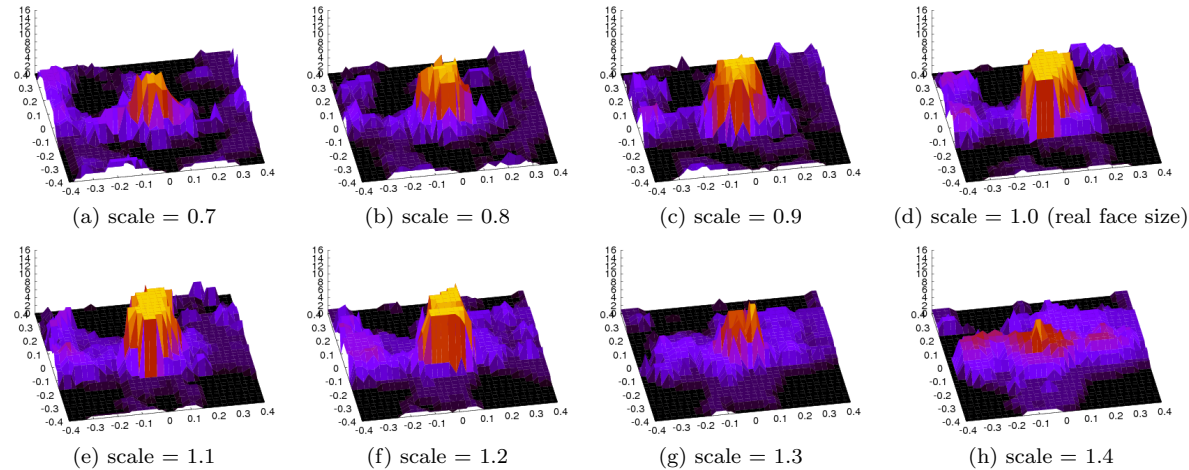
Fig. 2 Face likelihood distribution for change in x-axis and y-axis.

## 2.2 Face Likelihood Gained from Cascade Classifier

Many classifiers are used for difficult sub-windows that are very similar to a face in face classification, but only a few classifiers are used for easy sub-windows, which are obviously different from a face. This means that the number of layers outputting a “true” indicates the likelihood of a face in the input sub-window. In this paper, that number is defined as the “face likelihood”. A face likelihood is related to the face position and scale<sup>8)</sup>. In particular, not only does a sub-window for a correct face position have the maximum face likelihood (all the cascade layers output a true), but also the windows around the face position have a high face likelihood. In Ref. 8), the difference between the distribution around a real face position and a false-positive one is used for reducing the number of false positives. By using a face likelihood distribution for the face classification, the face detection accuracy is improved. In contrast, to speed up the face detection process, in the present study, we focus on the spread of the face-likelihood distribution based on the position and scale. We believe that by using the distribution the number of search points on which a face classification is performed can be cut. Accordingly, we examined the distribution according to the change in x-y position and scale.

## 2.3 Face Likelihood Distribution Dependence on Position and Scale

First, the face-likelihood distribution at the x-y positions is determined by using a sample cascade classifier. **Figure 2** shows a sample face image and the distribution gained from it. In the face-likelihood distribution graph, the x and y axes correspond to the axes on the sample image. Each value on the axes is shown as a normalized distance calculated by normalizing the distance from the



**Fig. 3** Face likelihood distribution for change in detector window size.

center of a face position based on the size of the face. The z-axis represents the face likelihood at each position. Note that the face-likelihood distribution is in the shape of gentle mountains in both the x and y directions. To put that concretely, the likelihood is the maximum value in the region within the distance of  $\pm 0.05$  [pixel/face size] and the low values in the region within a distance of  $\pm 0.1$ . The distribution is widely spread around the center position of the face. Second, the change in the face-likelihood distribution using the window scale of the detector is determined in the same way as described above (Fig. 2). The scale of the detector is increased from 0.7 to 1.4 magnifications at intervals of 0.1 according to the real-face scale = 1.0. **Figure 3** (a) to (h) show the distributions on each detector scale. It appears that the distribution is widest for a scale of 1.0 and gradually gets lower when the difference between the scale of the face and the detectors increases. However, it is clear that some points have a maximum likelihood on the distributions for a scale from 0.9 to 1.2 magnifications, and low and wide distributions are shown for scales from 0.7 to 1.3. It is thus concluded that the face likelihood is widely distributed for the three parameters, namely, the x position, y position, and scale. In particular, a low likelihood is widely distributed around the positions of a real face.

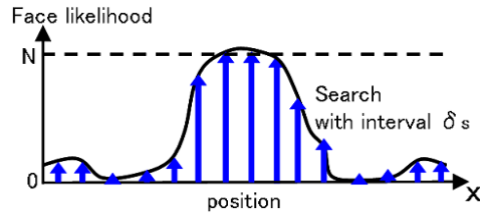
## 2.4 Fast Method of Face Detection Using Face Likelihood Distribution

This section focuses on the two characteristics of the cascade classifier. First, the number of cascade layers outputting a “true” equals face likelihood. Second, the face likelihood is distributed widely in relation to the face position and scale. The first characteristic contributes to the ease of operation for estimating possible face regions when using the initial layers in a cascade. The second reduces the number of search sub-windows needed. We investigated the proposed search method based on these two characteristics. The method is a combination of the face likelihood estimation using an easy operation and the search window reduction using the likelihood.

## 3. Proposed Method: Cascaded Step Search

### 3.1 Traditional Search Method

The traditional face-search method consists of a very simple procedure. It is a series of face classifications at multiple scales and positions at regular intervals (**Fig. 4**). First, in regards to the search scale  $s$ , a detector with a window size  $w_s$  scans over the entire image at regular intervals  $\delta_s$  so as to detect particularly



**Fig. 4** Image of traditional face search method. The horizontal axis indicates the  $x$  position in the input image, the vertical axis indicates the face likelihood at position  $x$ , and each up arrow indicates the face classification processes at each position.

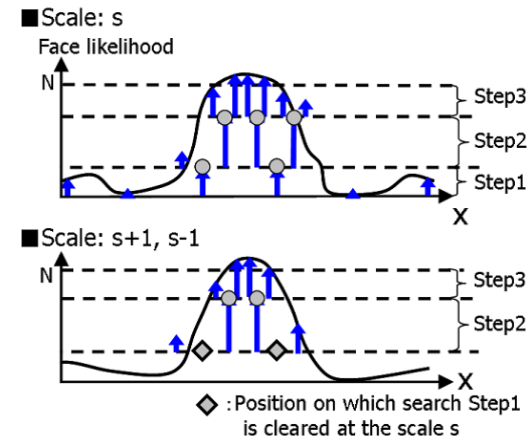
sized faces. Next, in regards to the search scale  $s + 1$ , the same search process is executed by using a detector with size  $w_{s+1}$  to detect different sized faces. The search parameters used in this search method are a detector size  $w_s$  and a position interval  $\delta_s$ . In regards to the detector scale  $s$ , the detector size is defined as  $w_s = w_0 \alpha^s$ , and the position interval is defined as  $\delta_s = \lceil \Delta * \alpha^s \rceil$ , where  $\lceil \cdot \rceil$  is a rounding operation,  $\alpha$  and  $\Delta$  are search parameters,  $\alpha$  is a detector scale factor,  $\Delta$  is a search position interval and  $s \geq 0$ . It was previously confirmed that the detection performance is best in terms of the accuracy and speed when  $\alpha = 1.25$  and  $\Delta = 1.5$  in Ref. 1).

A traditional search is a group of isolated classification processes. A narrow interval is thus required for a search so as not to miss the correct face position and scale, which are shown as a peak of the face likelihood in Fig. 4. The required interval is less than 10% of the detector width (Fig. 2). Consequently, the computational complexity is very high because many face classifications are carried out.

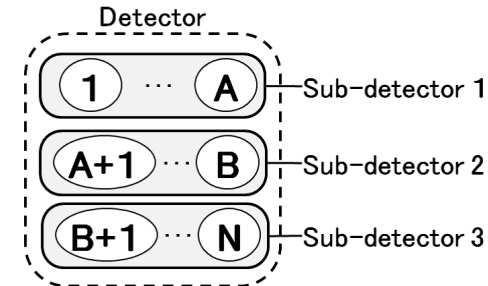
### 3.2 Proposed Search Method

#### 3.2.1 Concept

The proposed search method is an efficient search using two characteristics of a cascade classifier (which are described in Section 2.4). The method involves the three steps shown schematically in Fig. 5. Each step is performed with two ideas in mind, gradual classification and interval search, which are based on the characteristics. First, the gradual classification is based on the face likelihood distribution obtained by the classifier layers of the cascade. For the classification, the detector is divided into three sub-detectors which consist of sum layers of



**Fig. 5** Images of proposed face search method. This is drawn using the same format as in Fig. 4. The two graphs show each search process for scale =  $s$  and  $s + 1$ .



**Fig. 6** Sub-detector. Each circle indicates a layer of the cascade classifier, where  $1 \leq A < B < N$  and  $N$  is the total number of the layer.

the cascade (Fig. 6). The simple classification by using the sub-detector can determine stepwisely the face likelihood of a sub-window. So, the search method is divided into three steps by using each sub-detector. Second, the interval search can be performed by taking the face likelihood distribution into consideration. Using this distribution makes it possible to reduce the number of search points in terms of their position and scale. These easy operations can estimate the face likelihood distribution and detect a face. As a result of our consideration for

change of face likelihood distribution in Fig. 3, we designed below three steps.

### Step 1 Estimating Face Position

Step 1 is executed at wide intervals by using the sub-detector 1 with scale  $s$ . The search concept is shown in the upper graph of Fig. 5. Step 1 is not executed on scale  $s + 1$  and  $s - 1$ . The aim of this search is to estimate the base of a mountain-shaped face-likelihood distribution. The base is wide. Accordingly, this step can be simplified as the combination of a wide interval search and an easy classification by using the sub-detector 1 which has omitted classifiers. As a result, the possible face regions can be estimated by using this step.

### Step 2 Estimating Face Scale

For scales  $s$ ,  $s - 1$  and  $s + 1$ , Step 2 is executed around the positions that were cleared on the Step 1 classifications with scale  $s$ . The sub-detector 2 is used in this Step. This step is based on the change of face likelihood distribution according to detector window size, shown in Fig. 3. The possible face position gained in the Step 1 indicates the real face could be at the same positions in the scales of  $s - 1$ ,  $s$  and  $s + 1$ . So, this step is executed for three sequential scales by using sub-detector 2 which can respond narrower scales of real face. As a result, the possible face scale can be estimated by using this step.

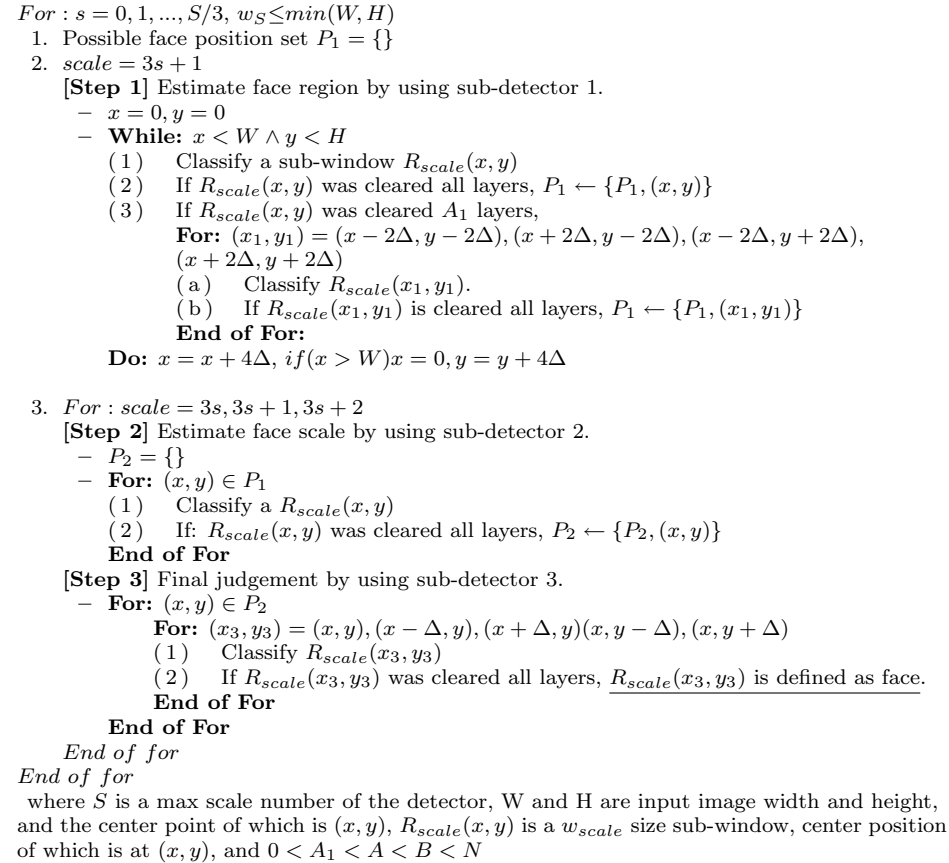
### Step 3 Final Judgment

A face search is executed on the positions covered by Step 2 at narrow intervals by using the sub-detector 3. Individual search positions are used for each scale in Step 3. In this step, the face positions at the top of the distribution are found by using a traditional interval search.

We aimed at reducing the cost of the non-face regions when performing search Steps 1 and 2. The proposed method is called a ‘‘Cascade Step Search’’, and it uses two ideas, gradual classification and interval search, which are based on the characteristics of a cascade classifier.

#### 3.2.2 Details of CSS Method

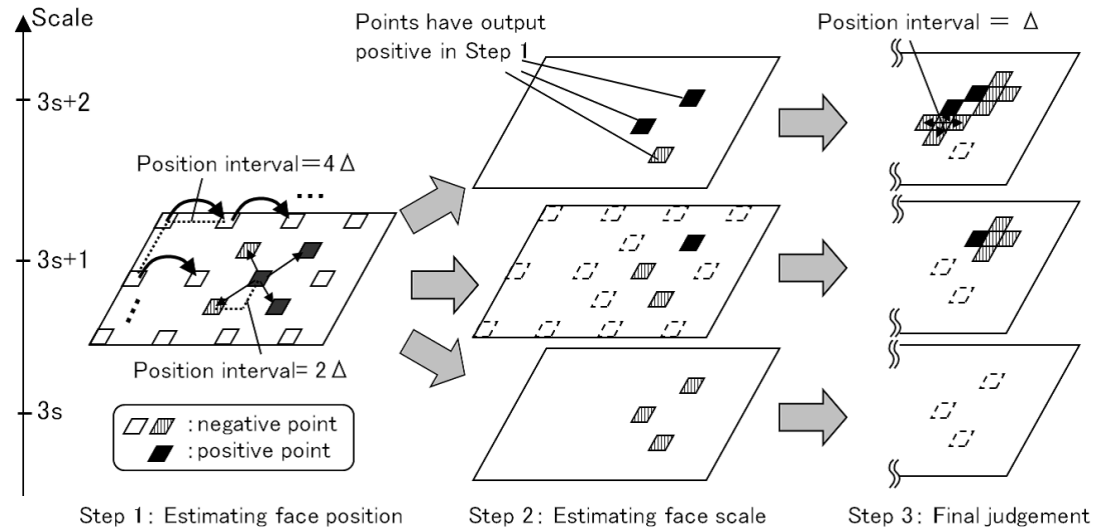
The CSS method described in **Figs. 7** and **8** shows an outline of the image processing. Step 1 targets only the center scale of every three scales, and the number of target scales is represented as  $3s + 1$ . It is composed of two sub-steps for searching the basis of the face-likelihood distribution. In the first sub-



**Fig. 7** Detailed flow of CSS method.

step, a sub-detector 1 scans an image at  $4\Delta$  intervals (Fig. 7 2.(1)–(2)). In the second sub-step, the target is the positive points from the first step and their circumferences (Fig. 7 2.(3)). As a result, possible face position set  $P_1$  is gained.

Step 2 for all scales starts from the same possible face points of Step 1. This step use a sub-detector 2 which react to a real face scale. The classifications are executed in each scale  $3s$  to  $3s + 2$ . As a result, only one of two scales clear this



**Fig. 8** Detailed processing image in CSS method. A negative point indicates the position of a sub-window that outputs a negative as a result of the face classification process in each step, and a positive point indicates the position of a sub-window that outputs a positive.

step, and they are classified as possible face scale. In Fig. 6, sample possible face scales are  $3s + 2$  and  $3s + 1$ . The possible face positions set  $P_2$  is gained in each scale.

Finally, a sub-detector 3 scans at all the positions in  $P_2$  and the neighbor. This is the final judgement which is face/non-face.

The search-position interval  $\delta$  in each step is controlled by the interval parameter  $\Delta$  of Step 3. As each search step is executed, the interval from  $4\Delta$  to  $\Delta$  narrows. Scale factor  $\alpha$  is 1.2, which is defined by the distribution change for scales shown in Fig. 3. Layer numbers  $A_1$ ,  $A$ , and  $B$  for each step are defined manually and heuristically by the shape of the face likelihood (Figs. 2 and 3), because they are changed by the training parameter of each cascade layer. While our method based on the above procedure can reduce the large number of search sub-windows, a fine search with a narrow interval is executed around the face point.

## 4. Experiments

### 4.1 Comparison Methods

We compared our method with some traditional search methods. To evaluate the efficiency of our method, traditional methods with three parameters were tested. These four test search methods are described below.

- (1) Traditional Search  
The traditional search is equivalent to the viola<sup>1)</sup> search method, which scans an input image at regular intervals for the scale and the position. As search parameters, a scale factor  $\alpha$  of 1.25 and a position interval  $\Delta$  of 1.0 were used. In all the tests, the detector base window size was 20x20 pixels. Therefore, in the first scale search, all the pixels are scanned, and a 25x25 pixel second-scale detector scans an image at a 1.25 pixel interval.
- (2) Wide Distance Search  
This traditional search is executed under the condition that the position

**Table 1** Results of each face-search method when false positive number is 100.

	1. Traditional	2. Wide Distance	3. Wide Scale	4. CSS
number of classifiers [num]	878M	220M	642M	109M
operating time [ms]	61.4	22.4	46.6	11.3
detection rate [%]	84.3	79.0	77.1	86.4

interval parameter  $\Delta = 2.0$ .

(3) Wide Scaling Search

This traditional search is executed under the condition that scale interval parameter  $\alpha = 1.4$ , which is the maximum interval for capturing the likelihood distribution estimated from Fig. 3.

(4) Cascade Step Search

The CSS is executed under the condition that search parameters  $\alpha$  and  $\Delta$  are the same as that of a Traditional Search.

The same cascade classifier was used as the detector for all four search methods. AdaBoost was used to train each layer of the cascade using 14,000 positive and 7,000 negative image samples. The positive and false images are collected by downloading images on the world wide web and taking face and non-face pictures. The total number of layers,  $N$ , was 20. Two experiments were executed. First, we tested all the search method on MIT+CMU frontal face test set for performance test. Second, we tested two methods which are traditional and the CSS, so as to evaluate the robustness for face variation.

#### 4.2 Performance Test

First, we tested all the search methods on the MIT+CMU frontal face test set (including 117 images with 507 frontal faces) for performance test. The detection rate and calculation cost were used as benchmarks for comparison of the CSS and the traditional methods. Two values were used to evaluate the cost, the operating time (OT), which is obtained as a sum total for all the test images for a PentiumD 3.4GHz CPU, and the number of weak classifications (NC), which are obtained as the sum of all face classifications with weak classifiers from all the image searches.

**Table 1** shows the results of using all the search methods on the test set. Our search method can reduce the NC for all the classifications to 109 million, which

**Table 2** Change of accuracy (detection rate / false positive number) for cascade layers in each face-search method.

Layer	1. Traditional	2. Wide Distance	3. Wide Scale	4. CSS
18	87.2% / 187	77.3% / 61	81.7% / 134	86.4% / 100
19	85.4% / 128	74.0% / 40	77.1% / 97	83.2% / 64
20	83.2% / 76	70.2% / 28	73.4% / 59	81.7% / 37

is 12.5% of that of the traditional method, i.e., 878 million. This means that the OT of the CSS is about 18.4% faster. The difference between the reduction rates of NC and OT is due to the scanning cost and multiple initializing, because the CSS method requires three scans per image at each search step. Furthermore, the CSS method can keep the face detection rate the same as that of the traditional methods. On the other hand, by simply reducing the number of search points, search methods 2 and 3 can reduce the rate by more than 5%, but their speed-up rate is less than that of the CSS method. It is thus concluded that our CSS method can reduce only the waste processing on the non-face region, and accomplishes about 5 times faster method of face detection while maintaining a high detection rate.

**Table 2** shows the difference of detection rate and false positives according to the layer number of the cascade classifier. It indicates that the detection rate and the false alarm numbers in 2-4 methods are lower than the traditional one, because their method skips many computations. On the other hand, only the CSS method can keep the detection rate within a 2% decrease, and reduce about a half false positives. The CSS's detection rate is higher than traditional one on the condition that all methods' false numbers are the same. It indicates that correct search based on the face likelihood can reduce the processing time and many false positives.

The change in the face detection rate with false positives is plotted as ROC

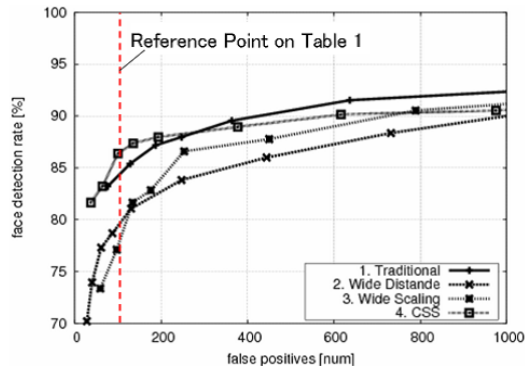


Fig. 9 ROC curves for comparison on CMU-MIT testing sets.

curves in Fig. 9. The lines for search methods 2 and 3 drop at a steeper rate with a decreasing number of false positives. The drops are affected by the shape of the face-likelihood distribution. The distribution region gradually narrows when the likelihood increases. Since a simple, wide-interval search such as these methods miss the top region of the distribution, the detection rate falls. In contrast, the line for the CSS method keeps the face detection rate relatively high. This result indicates that our method can find face positions correctly by using the face-likelihood distribution obtained from a cascade classifier.

### 4.3 Face Variation Test

Second, we tested only two search methods, which are traditional and proposed, the test set had two particular face variations. The variations are the number of faces in an image and the face size. In the proposed method, the number affects the operating time because it uses the cascade classifier. The size affects the detection rate because the search process on each scale is varied. This test intends to evaluate the robustness of the proposed method. Two test image sets were made manually for the experiment. They have different face size, one is 36x36 pixels face and another is 28x28 pixels, in QVGA image. the former is one of Step 1 search size in the proposed method (scale = 3s + 1 in Fig. 7) and the latter is the size that has the largest distance from the two Step 1 size (are 36x36 and 20x20). It tends to be that the former size face is detected accurately but the latter size is inaccurately. All faces are picked up and rescaled from the BANCA

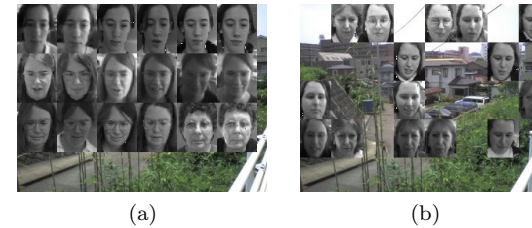


Fig. 10 Variation test set image. (a) is a 36x36 test set image sample which has 18 faces. (b) is a 28x28 test set image sample which has 15 faces.

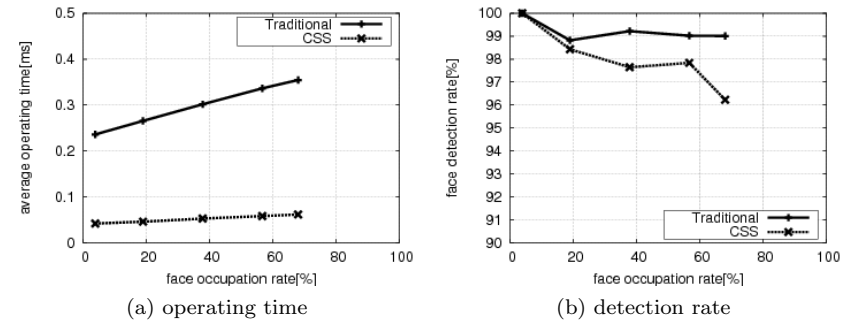


Fig. 11 Performance variation according to the face occupation rate in an image on 36x36 face size test set.

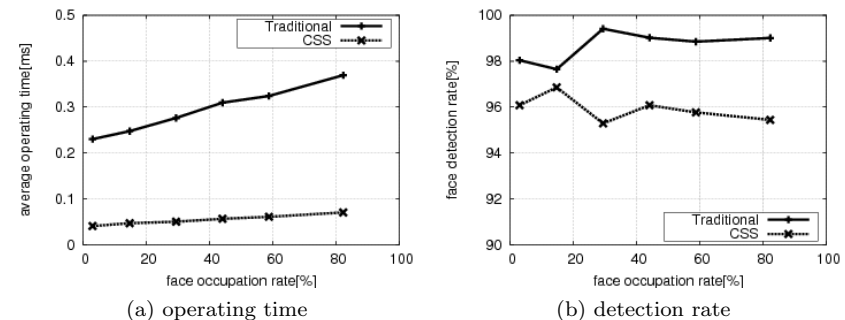
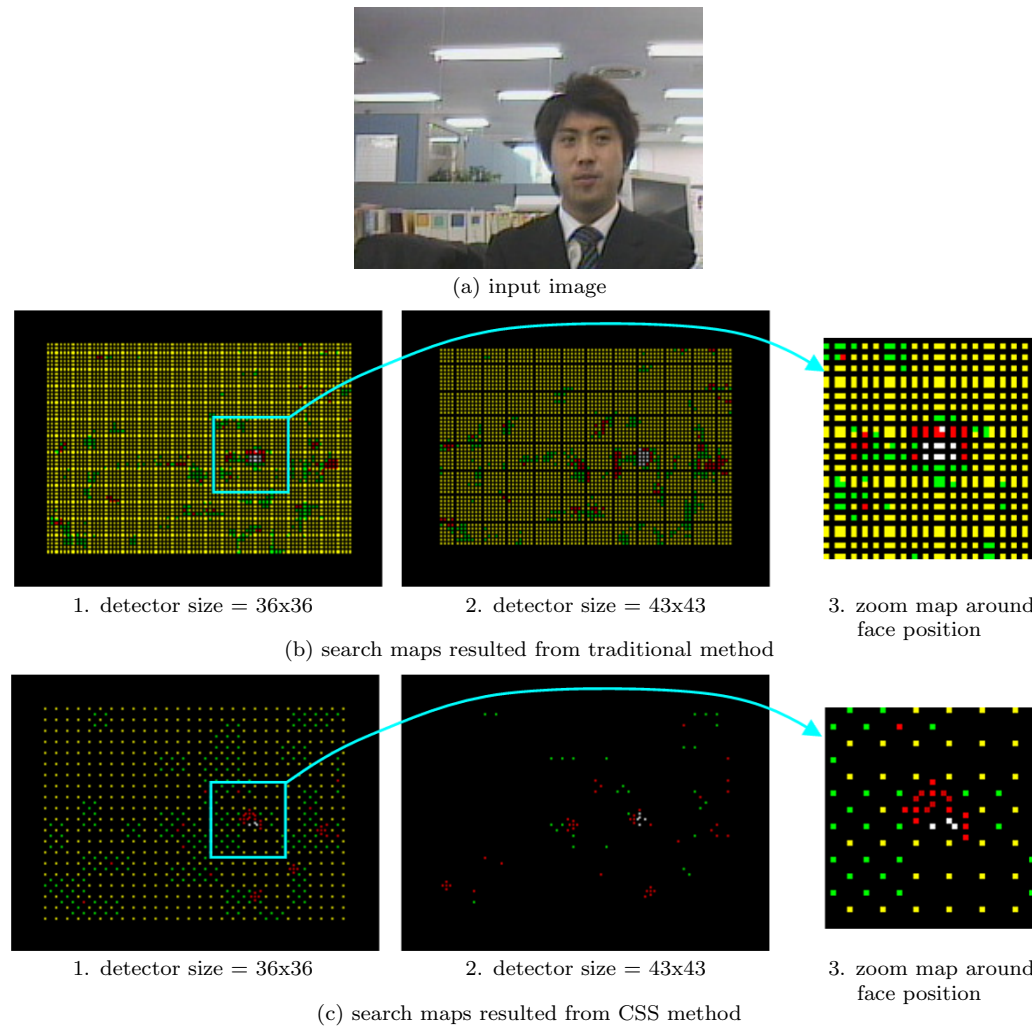


Fig. 12 Performance variation according to the face occupation rate in an image on 28x28 face size test set.





**Fig. 13** Search image maps resulting from two search methods: (a) shows an input image, (b) maps showing results of the traditional search method in an image (a). Each point shows whether face classification was executed or not, and their color is the output value of the classification. (1) Result of search with detector with a 36x36 pixel window size, (2) is that with a 43x43 pixel size and (3) shows a zoom map that picked up from (1) around the face. (c) Maps showing results of CSS method. The colored points indicate the face likelihood: each yellow point indicates the Step 1 searched point, green indicates Step 2, red indicates Step 3, and white indicates the judged face point.

database<sup>12)</sup>, and are set on the same background image. Each test set consists of sub-sets, each of which include 50 images. All images in the sub-set have the same number of different faces, and the number is different in each subset. The number of faces is 1, 5, 10, 15, 18 in the 36x36 size test set, and the number is 1, 5, 10, 15, 20, 28 in 28x28 size test set. The sample images of the test sets are shown in **Fig. 10**. The two search methods were tested on the test sets, and the average operating time per image and the detection rate are evaluated.

**Figures 11 and 12** show the results on the 36x36 and 28x28 size face test sets. In these figures, “the face number in a image” is modified to a “face occupation rate in an image”. The operating time curves of all methods on all test sets are in proportion of the face occupation rate. The operation time reduction rate between the traditional and the CSS methods keeps about five times, i.e., 0.35 and 0.061 ms in the number of 18 faces sub set on the 36x36 size set. This result indicates that the time reduction rate on positive and negative images are the same. The detection rate varies a little from the face occupation rate and the face size. In 36x36 face test set, the difference in the detection rate between the traditional and the CSS method is about 1%. On the other hand, the difference is about 3% in 28x28 face test set. This result indicates that the proposed method keeps the detection rate within a 3% decrease.

## 5. Discussion

We have investigated the search process by outputting points on which a face classification has been executed and the likelihood gained as a result of the classification. The results are shown as maps in **Fig. 13**. As for the traditional method, the search points are distributed at regular intervals (Fig. 13 (b)). In contrast, for the CSS method, the points are widely and randomly distributed (Fig. 13 (c)). It is clear that the number of search points in the CSS method is reduced. In the search using a 43x43 detector (Fig. 13 (b)-2 and (c)-2), the large number of search points is reduced by CSS Step 2 (face scale estimation), but the positions on which the face likelihood is high are correctly searched. Figure 13 (b)-3 and (c)-3 show the search points around a face. The search points in the CSS method are closely distributed on the points around a face. So, CSS is executed closely on the region around a face, although they are widely executed

on a non-face region. So, the CSS method can perform a faster face detection (namely, more than five times faster) while maintaining a detection rate.

## 6. Summary

We proposed a fast CSS method of face detection. The proposed method reduces the operating time by conducting an efficient search that is based on two features. First, a gradual classification that uses only a few layers of a cascade classifier to estimate the face likelihood distribution. Second, an interval search that uses the face likelihood distribution. The search is operated at an interval that is optimally changed according to the likelihood. The CSS reduces the number of sub-windows that must be processed. These features can reduce the face classification on the waste position, which is the non-face region, and closely and correctly searches the face region. Our experiments on face detection show that the proposed method is about five times faster than traditional searches and maintains a high detection rate.

**Acknowledgments** The research on which this paper is based acknowledges the use of the Extended Multimodal Face Database and associated documentation. Further details of this software can be found in; K. Messer, J. Matas, J. Kittler, J. Luetin and G. Maitre; “XM2VTSdb: The Extended M2VTS Database, Proceedings 2nd Conference on Audio and Video-base Biometric Personal Verification (AVBPA99)” Springer Verlag, New York, 1999. CVSSP URL: <http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb>.

## References

- 1) Viola, P. and Jones, M.: Neural network-based face detection, Rapid Object Detection using a Boosted Cascade of Simple Features, *Proc. CVPR*, Vol.1, pp.511–518 (2001).
- 2) Lienhart, R., Kuranov, A. and Pisarevsky, V.: Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection, *Proc. German 25th Pattern Recognition Symposium (DAGM'03)* (2003).
- 3) Mita, T., Kaneko, T. and Hori, O.: Joint Haar-like Features Base on Feature Co-occurrence for Face Detection, *IEICE Trans. Inf. Syst.*, D, pp.1791–1801 (2006).
- 4) Wu, B., Ai, H., Huang, C. and Lao, S.: Fast Rotation Invariant Multi-View Face Detection Based on Real Adaboost, *FG2004*, pp.79–84 (2004).
- 5) Rowley, H., Baluja, S. and Kanade, T.: Neural Network-based face detection,

Pattern Analysis and Machine Intelligence, *IEEE Trans. Pattern Analysis*, Vol.20, pp.23–38 (1998).

- 6) Schapire, R.E., Freund, Y., Bartlett, P. and Lee, W.S.: Boosting the Margin: A new explanation for the effectiveness of voting methods, *The Annals of Statistics*, Vol.26, No.5, pp.1651–1686 (1998).
- 7) McCane, B., Novins, K. and Albert, M.: Optimizing Cascade Classifiers, [http://reflect.otago.ac.nz/staffpriv/mccane/publications/optimising\\_cascades.pdf](http://reflect.otago.ac.nz/staffpriv/mccane/publications/optimising_cascades.pdf) (2005).
- 8) Takatsuka, H., Tanaka, M. and Okutomi, M.: Face Detection Based on the Distribution of Classifier Outputs, *IPSP Trans. Computer Vision and Image Media (CVIM19)*, Vol.48, No.SIG\_16, pp.51–54 (2007).
- 9) Po, L. and Ma, W.: A Novel Four-Step Search Algorithm for Fast Block Motion Estimation, *IEEE Trans. Circuits Systems Video Technol.*, pp.313–317 (2002).
- 10) Murase, H. and Vinod, V.V.: Fast visual search using focused color matching — active search, *System and Computers in Japan*, Vol.31, No.9, pp.81–88 (2000).
- 11) Fleuret, F. and Geman, D.: Coarse-to-Fine Face Detection, *International Journal of Computer Vision*, Vol.41, pp.85–107 (2001).
- 12) Bailly-Bailliere, E., Bengio, S., Bimbot, F., Hamouz, M., Kittler, J., Mariethoz, J., Matas, J., Messer, K., Popovici, V., Poree, F., Ruiz, B. and Thiran, J.P.: The BANCA database and evaluation protocol, *Proc. 4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)* (2003).

(Received May 7, 2010)

(Accepted January 26, 2011)

(Released May 11, 2011)

(Communicated by *Chu-Song Chen*)



**Tomoaki Yoshinaga** received his Master of Computer Sciences from Tokyo Institute of Technology in 2004, and was engaged in research on image pattern recognition with Hitachi Ltd. from 2004. His research interests include human image recognition, multi-modal recognition and machine learning. He is a member of IEICE.



**Shigeki Nagaya** received his Master of Engineering from Sophia University in 1992. He joined the Hitachi Central Research Laboratory in 1992 and resigned in 2010. His research interests include gesture motion image understanding, multi-modal man-machine interface, video surveillance and machine learning.



**Isao Karube** received his B.E., M.E. degrees in Electronic Engineering from the University of Tokyo in 2000 and 2002, respectively. He joined Hitachi, Ltd. in 2002, and has been engaged in research and development of video coding. His current research interests include video coding and learning algorithms.