15

Bayesian Video Face Detection with Applications in Broadcasting

Atsushi Matsui [1], Simon Clippingdale [1], Norifumi Okabe [2], Takashi Matsumoto [3] and Nobuyuki Yagi [1]

[1] *Science & Technology Research Laboratories, Japan Broadcasting Corporation, Tokyo, Japan*
[2] *Broadcast Engineering Department, Japan Broadcasting Corporation, Tokyo, Japan*
[3] *Faculty of Advanced Science and Engineering, Waseda University, Tokyo, Japan*

1. Introduction

Face detection is the most fundamental and critical process in any automated system that deals with face images. Errors in this first step affect subsequent processes such as face localization, face modeling, facial expression analysis, face recognition, user classification, and so on. A face detector should have sufficient robustness to possible variations of face position, scale, orientation, aging, make-up, and illumination.

Several approaches based on different features are available for face detection: model-based, color-based, appearance-based, or a combination of these. Model based approaches can deal with variations in face pose and illumination, but require the initial position of the target a priori. Color based approaches can reduce the search space of the detection system. However, skin color models are not effective where the spectrum of the light source varies significantly, i.e. color appearance is often unstable due to changes in both background and foreground lighting. To the best of the authors' knowledge, the most successful face detection algorithms are based on appearance without using other cues. Although there has been much reported research in this field, it is probably fair to say that the framework of Viola and Jones (Viola & Jones, 2001) has attracted the most attention for its combination of detection accuracy and speed. They introduced a cascaded structure of weak classifiers using a boosted learning algorithm on a pool of simple Haar-like features (Papageorgiou, et al., 1998), and an extremely fast method of evaluating the features at any image location and scale. Lienhart and Maydt (Lienhart & Maydt, 2002) demonstrated the efficacy of extending the feature set; the modified version of the Haar-cascade face detector is available in the open-source computer vision library OpenCV. The computational processes for the face detection system can be divided into two stages: the learning stage and the detection stage. The learning stage involves the selection of the feature pool, variation of training samples, and training algorithm. The detection stage includes the search algorithm, structure, and merging post-processing of multiple outputs.

In this paper, we introduce an online learning scheme based on a Sequential Monte Carlo method (Doucet, 1998); (Freund & Schapire, 1997) to boost computational efficiency on the detection stage.

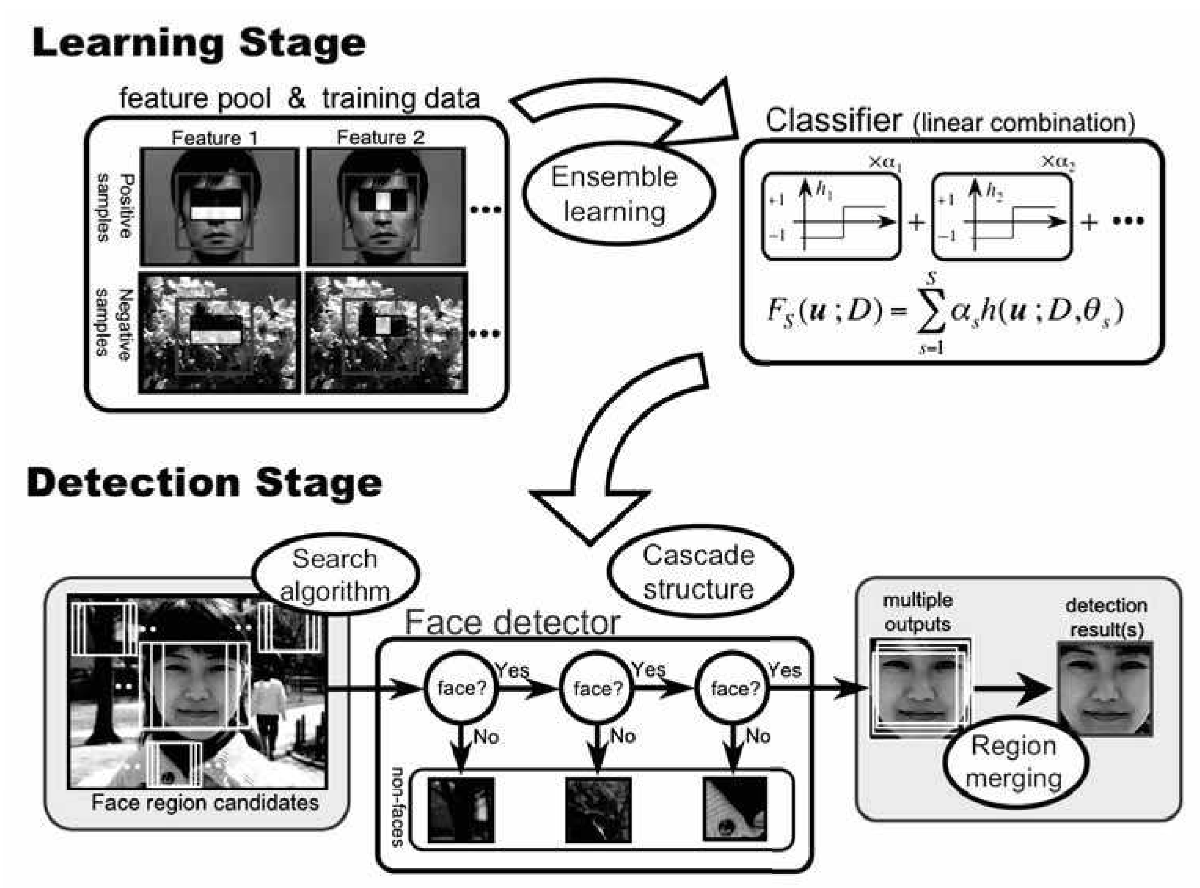

Fig. 1. An overview of a Haar-cascade face detector and associated processes.

2. Bayesian approaches for detection stage

As in numerous other pattern classification tasks, accuracy and speed are twin requirements of face detection systems, with speed especially vital for some video-based applications. More accuracy, however, often requires more computation and thus entails less speed. A combination of a face detector and a tracker (to reduce the search volume in the current frame based on results up to the previous frame) may be used to increase processing speed in such cases. However, the underlying assumption of temporal continuity is violated at scene changes, which must be detected and the system re-initialized in order to avoid generating erroneous detections. In the following sections, we introduce a sequential face detection scheme for video sequences containing scene changes. The algorithm estimates probability distributions of a sequence of face region parameters using a Sequential Monte Carlo (SMC) method (Doucet, 1998). We show that this SMC approach successfully predicts possible regions in face parameter space for the current frame using temporal continuity from past frames, while resetting the distributions automatically at temporal discontinuities corresponding to scene changes.

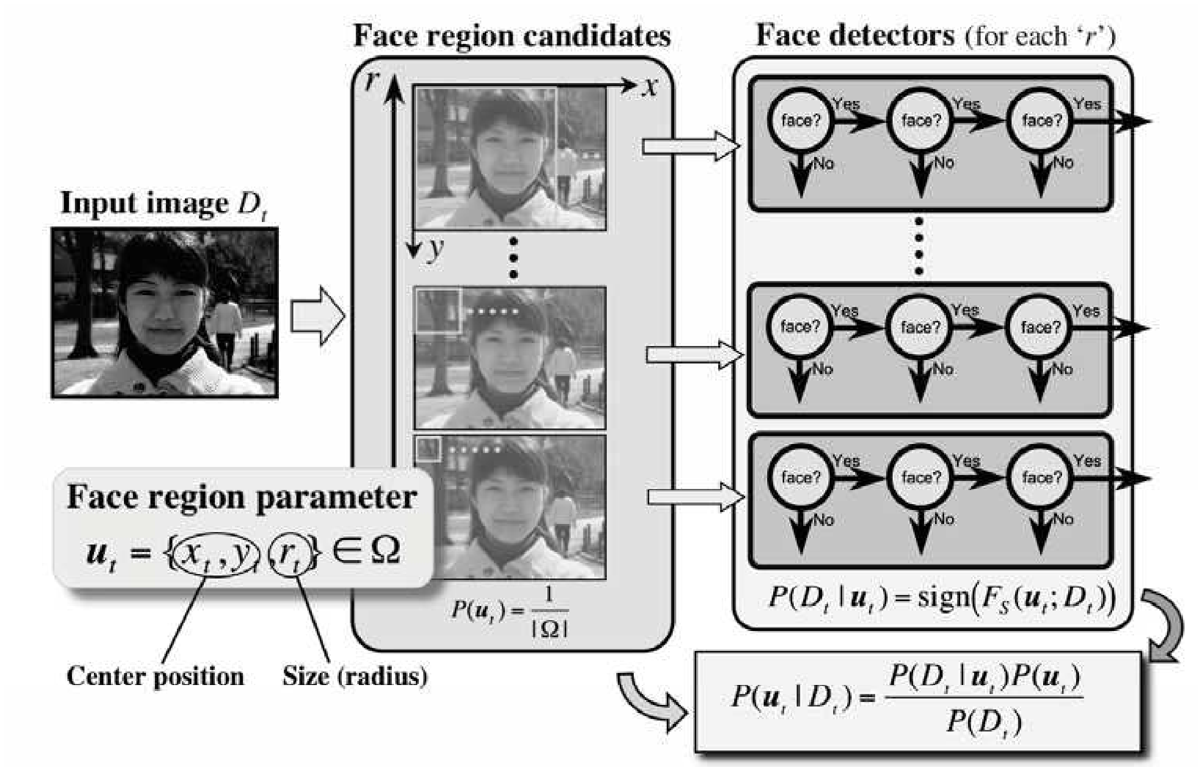

Fig. 2. Overview of an exhaustive search algorithm.

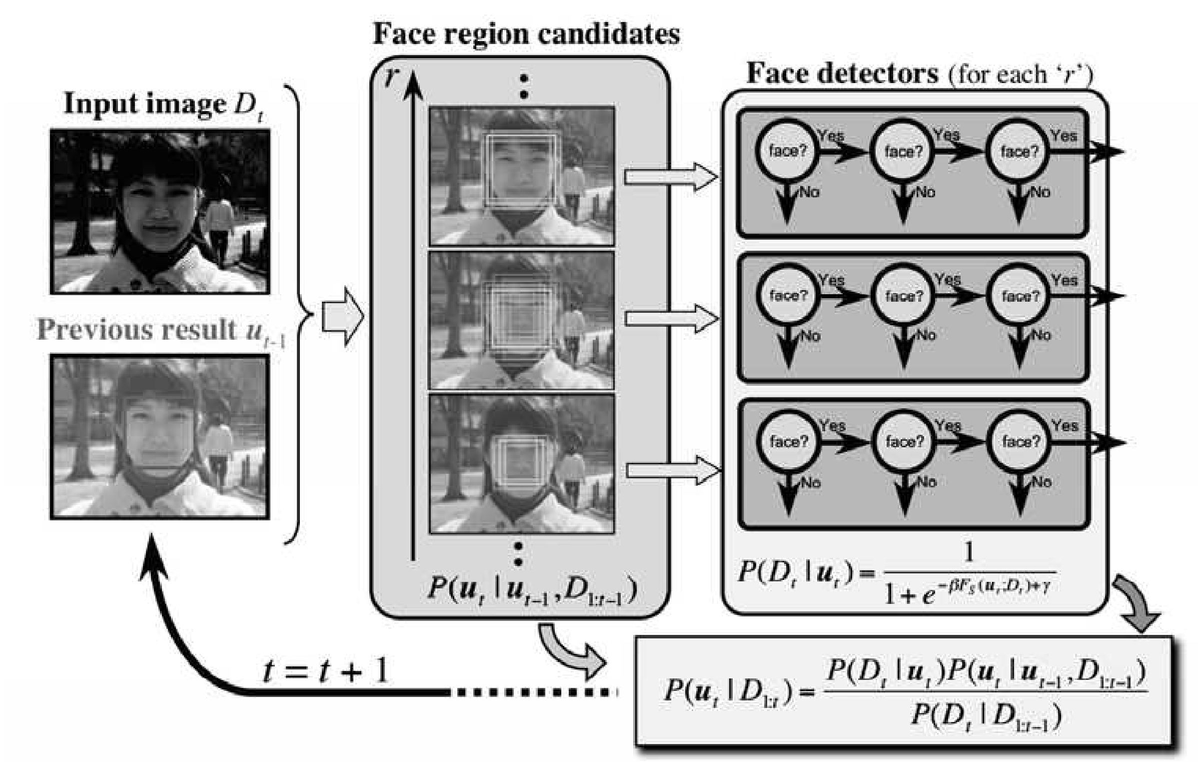

Fig. 3. Overview of the Bayesian sequential importance search algorithm.

2.1 Exhaustive search and frequency-based thresholding

Given an input image, the face detector scans possible combinations of the center positions and sizes of face rectangles. The computational cost reflects the total number of face candidates. Such face detectors are often trained using a training data set with some variations in position and scale, and the prevailing features for each weak learner are relatively simple and have a degree of uncertainty such that a single face in an input image can yield multiple face candidates with a range of positions and scales. One possible approach to handling these multiple outputs is to classify neighboring candidates into groups G_j , and to take an average over all elements of each group(see figure 1).

$$\hat{\boldsymbol{u}}^{[j]} = \frac{1}{|G_j|} \sum_{\boldsymbol{u}_i \in G_j} \boldsymbol{u}_i \ , \quad \boldsymbol{u}_i ; \ H_S(\boldsymbol{u}_i ; D_t) = 1 \tag{1}$$

Since true positives usually occur with greater consistency than false positives, if the number of elements belonging to the group G exceeds some threshold η the group may be considered likely to represent a true positive, while groups with fewer members may be rejected on the grounds that they are more likely to represent false positives.

2.2 Bayesian importance search and marginalization

In the general case, a face detection system is designed to scan the entire possible space of face region parameters uniformly unless we have prior knowledge of the statistics of those face parameters in the input image. It can be regarded as a maximization of the posterior distribution of the parameters using the uniform prior distribution (see figure 2). However, in the case of video face detection, we can assume temporal continuity of face regions as well as that of input image frames in general (see figure 3). Avidan (Avidan, 2001) introduced the first combined approach which uses the output of an SVM object detector to perform a tracking task. Okuma (Okuma, et al., 2004) proposed an SMC based algorithm which combined the AdaBoost detector and a color-based object tracker. The system can successfully track multiple targets from a given video sequence, but does not use the classifier directly as a likelihood function.

Consider the situation where data is given as a video sequence. Let D be image data at the current (t^{th}) frame and let $D_{1:n} = \{D_1, D_2, ..., D_n\}$ be the image data set up to the current frame. Several approaches regard the output of the face classifier $F_S(\boldsymbol{u}_t ; D_t)$ as a confidence score for a face candidate $\boldsymbol{u}_t$. The likelihood function in this work is defined by using the calibration technique of Platt's scaling (Platt, 1999)

$$P(D_t | \boldsymbol{u}_t) = \frac{1}{1 + \exp\left(-\beta F_S(\boldsymbol{u}_t ; D_t) + \gamma\right)} \tag{2}$$

Using Bayes' theorem we obtain straightforwardly a recursive formula for $P(\boldsymbol{u}_{0:t}|D_{1:t})$, the joint posterior distribution of a series of face region parameters $\boldsymbol{u}_{0:n} = \{ \boldsymbol{u}_0, \boldsymbol{u}_1, ..., \boldsymbol{u}_n \}$ given the data:

$$P(\boldsymbol{u}_{0:t} \mid D_{1:t}) = \frac{P(D_t \mid \boldsymbol{u}_t) P(\boldsymbol{u}_t \mid \boldsymbol{u}_{t-1}, D_{1:t-1})}{P(D_t \mid D_{1:t-1})} P(\boldsymbol{u}_{0:t-1} \mid D_{1:t-1}) \tag{3}$$

with $P(D_t | D_{t-1}) = P(D_1)$ at $t = 1$. Consider a *proposal distribution* π such that draws

$$\boldsymbol{u}_{0:t}^{(i)} \sim \pi(\boldsymbol{u}_{0:t} \mid D_{1:t-1}), \quad i = 1,\ldots,N \tag{4}$$

can be obtained by standard methods, and define the *importance weights*

$$w_{0:t}^{(i)} = \frac{P(D_{1:t} \mid \boldsymbol{u}_{0:t}^{(i)}) P(\boldsymbol{u}_{0:t}^{(i)} \mid D_{1:t-1})}{\pi(\boldsymbol{u}_{0:t}^{(i)} \mid D_{1:t-1})}, \quad i = 1,\ldots,N \tag{5}$$

Then Sequential Monte Carlo approximates the joint posterior distribution by

$$\hat{P}(\boldsymbol{u}_{0:t} \mid D_{1:t-1}) = \sum_{i=1}^{N} \tilde{w}_{0:t}^{(i)} \delta\left(\left\| \boldsymbol{u}_{0:t} - \boldsymbol{u}_{0:t}^{(i)} \right\|\right) \tag{6}$$

$$\tilde{w}_{0:t}^{(i)} := \frac{w_{0:t}^{(i)}}{\sum_{j=1}^{N} w_{0:t}^{(j)}} \tag{7}$$

In order to perform Monte Carlo evaluation of the one-step marginal likelihood, assume that the draws at the previous step

$$\boldsymbol{u}_{0:t}^{(i)} \sim P(\boldsymbol{u}_{0:t-1} \mid D_{1:t-1}), \quad i = 1,\ldots,N \tag{8}$$

are available. Use a stochastic dynamics for the parameters $P(\boldsymbol{u}_t \mid \boldsymbol{u}_{t-1}, D_{1:t-1})$ to generate $\{\tilde{u}_t^{(i)}\}_{i=1}^{N}$ and evaluate

$$\hat{P}(D_t \mid D_{1:t-1}) = \sum_{i=1}^{N} P(D_t \mid \tilde{\boldsymbol{u}}_t^{(i)}) \tilde{w}_{0:t-1}^{(i)} \tag{9}$$

If we design a sequential proposal distribution of the form

$$\pi(\boldsymbol{u}_{0:t} \mid D_{1:t-1}) = \prod_{s=1}^{t} \pi(\boldsymbol{u}_s \mid \boldsymbol{u}_{0:s-1}) \pi(\boldsymbol{u}_0) \tag{10}$$

then, instead of drawing entire sample trajectories at each time by (8), we only need to draw one-step samples

$$\boldsymbol{u}_t^{(i)} \sim \pi(\boldsymbol{u}_{0:t} \mid \boldsymbol{u}_{0:t-1}^{(i)}, D_{1:t-1}) \tag{11}$$

which significantly reduces computational costs. A sequential proposal distribution also leads to sequential importance weights:

$$w_{0:t}^{(i)} = \frac{P(D_t \mid \boldsymbol{u}_t^{(i)}) P(\boldsymbol{u}_t^{(i)} \mid \boldsymbol{u}_{0:t-1}^{(i)}, D_{1:t-1})}{\pi(\boldsymbol{u}_t \mid \boldsymbol{u}_{0:t-1}^{(i)}, D_{1:t-1})} w_{0:t-1}^{(i)}, \quad i = 1,\ldots,N \tag{12}$$

We predict the region for the j^{th} face candidate group G_j by the marginal posterior mean of the parameters at time t :

$$\hat{\boldsymbol{u}}_t^{[j]} = \frac{\int_{\boldsymbol{u}_t \in G_j} \boldsymbol{u}_t P(\boldsymbol{u}_t \mid D_{1:t}) d\boldsymbol{u}_t}{P(D_t \mid D_{1:t-1})_{G_j}} \cong \frac{\sum_{\boldsymbol{u}_t \in G_j} \boldsymbol{u}_t^{(i)} \tilde{w}_{0:t}^{(i)}}{\sum_{\boldsymbol{u}_t \in G_j} \tilde{w}_{0:t}^{(i)}} \tag{13}$$

It should be noted that the Monte Carlo marginalization can be implemented simply by discarding those components that are not of interest. This is one of the advantages of Monte Carlo methods in general, and Sequential Monte Carlo in particular.

2.3 Change detection and re-initialization

In many applications including broadcasting, input video sequences can contain discontinuities corresponding to scene changes. Since the approach described above assumes continuity between the current video frame D_t and the past series of frames $D_{1:t-1}$, we need a change detection framework to re-initialize the probability distributions at scene changes.

We propose the *sequential marginal likelihood* (Matsumoto & Yosui, 2007) $P(D_t| D_{1:t-1})$ as representing the degree of continuity of the given sequence $D_{1:t}$. If the marginal likelihood suddenly drops, it is natural to suppose that an unexpected phenomenon on the input sequence, like a scene change, may have occurred. We therefore reset the probability distribution of $\boldsymbol{u}_t$ to beflat if the sequential marginal likelihood $P(D_t \mid D_{1:t-1})$ for all face candidate groups G falls below a threshold ε:

$$\pi(\boldsymbol{u}_t \mid \boldsymbol{u}_{0:t-1}^{(i)}, D_{1:t-1}) = \begin{cases} \mathcal{N}(\boldsymbol{u}_{t-1}^{(i)}, \Sigma) & \text{if } {}^{\exists}G_j; P(D_t \mid D_{1:t-1})_{G_j} > \varepsilon, \\ 1/|\Omega| & \text{otherwise,} \end{cases} \tag{14}$$

where Σ denotes the 3x3 covariance matrix of the Gaussian distribution $\mathcal{N}$ and Ω denotes the size of the entire parameter space of $\boldsymbol{u}_t$.

2.4 Experiments

We evaluated the performance of the proposed algorithms, compared with a face detection algorithm provided by the OpenCV libraries[11]. As a test image sequence, we selected 500 frames from a video sequence from the TRECVID 2007 development data, a collection of broadcast videos. The development data provides a wide variety of broadcast programs to train content-based retrieval systems for the TRECVID contest, at QVGA resolution (320x240) with the frame rate of the PAL video format (25Hz). We selected the "BG_15190" video sequence, which contains many frontal faces at relatively large sizes. We constructed the test sequence from the first 10 frames of each of the first 50 shots, giving 500 frames containing 450 faces. We entered ground truth data by hand for all frontal and semi-frontal faces. Evaluation of the detection results was performed against the ground truth data, allowing position and size errors of up to 10% of the true face size. To simply compare each search strategy in face region parameter space, we used the same core object detector function from OpenCV as the face classifier $F_s(\boldsymbol{u}_t ; D_t)$, and used the same runtime data file for frontal faces. For the baseline algorithm, we tried 10 settings for the threshold η (= {1, 2, . . . , 10}). For the proposed algorithm, we tried 10 settings for the threshold ε (= {0.2, 0.4, . . . , 2.0}), and fixed the number of Monte Carlo samples at $N = 500$.

Figure 4 and figure 5 show performance curves (precision vs recall, and F-measure vs detection speed) for the baseline algorithm and the proposed algorithm. Figure 4 shows that the average speed of the proposed approach was roughly double that of the baseline scheme for all settings, without sacrificing detection performance.

Table 1 shows details of the best results in terms of F-measure for each approach. Figure 5 shows temporal variations in detection time per frame for the proposed method with $\varepsilon = 1.2$. Spikes in the detection time are due to enlargement of the parameter search space corresponding to re-initialization when the lower alternative in equation (14) is selected. The frequency of such spikes on the trajectory corresponds to the 10-frame interval at which discontinuities occurred in the test sequence (constructed from 50 different shots of length 10 frames).

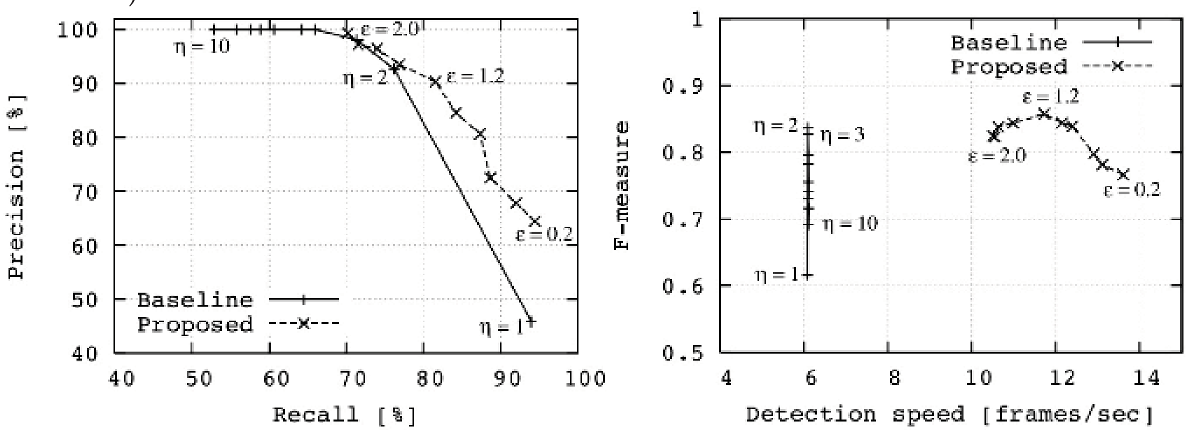

Fig. 4. Face detection performance of both methods with various thresholds (Left: precision vs recall, Right: F-measure vs detection speed).

Search scheme	Baseline (η = 2)	Proposed (ε = 1.2)
Recall	76.2 % (343/450)	81.6 % (367/450)
Precision	92.7 % (343/370)	90.4 % (367/406)
F-measure	0.84	0.86
frame rate	6.1 fps	11.7 fps

Table 1. Parameters at best (largest F-measure) result for each algorithm.

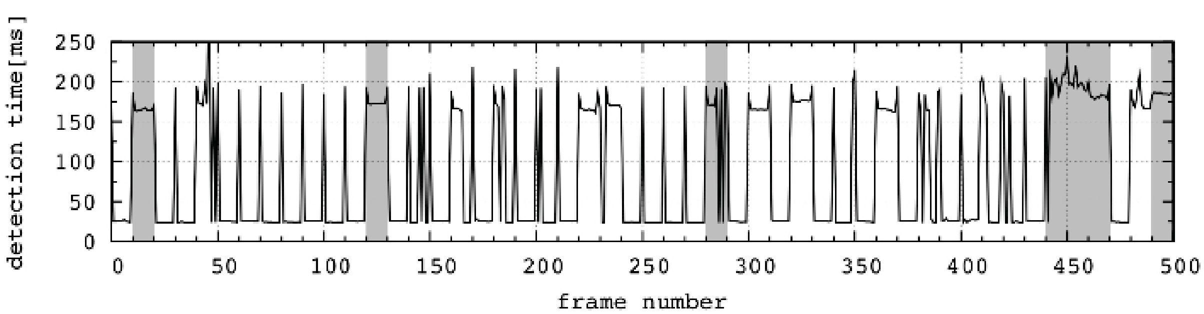

Fig. 5. Trajectory of detection time for proposed method (ε = 1.2), performed on a 3.8 GHz Xeon CPU. The grey regions indicate the input frame with no face.

3. Applications in Broadcasting

Many applications of face detection technologies have appeared in the last few years, and digital still camera featuring the technologies have achieved commercial success. However, there are still few applications for video sequences. In the following sections, we introduce two particular applications of video face detection in broadcasting.

3.1 Video Correction Support System

In this section, we describe a video correction system for broadcast video materials. The system automatically detects human faces in the input video sequence as target regions, then estimates an appropriate set of correction parameters for the white/black/gamma levels of the detected regions. Figure 6 shows an overview of the system, constructed from 5 modules: color corrector, 4-split multi viewer, face detection PC, correction control PC, and GUI terminal.

The 4-split multi viewer encodes the input video signal as a sequence of JPEG images and broadcasts them to the two PCs, locally connected with Ethernet cables. Then the face detection PC detects all possible face candidate regions and passes them to the correction control PC. For each detected face region, the correction control PC evaluates the average luminance, and estimates the optimum luminance transformation so as to achieve a natural face tone.

The correction control PC also evaluates a histogram of luminance over the whole input image, and sets black/white/gamma levels so as to achieve appropriate contrast and dynamic range for broadcasting. The set of video correction parameters are smoothed using a moving average filter, so as to eliminate over-sensitive responses to transient changes like flashing, and to eliminate possible errors in the face detection stage. Then the color corrector transforms the input video signal according to the set of smoothed correction parameters, as determined by the correction control PC. Using the GUI terminal (see figure 7), operator(s) can check a set of sample images from the input/output video signals. The GUI also provides adjustment/preset functions for each system parameter, including weighting factors for the whole image and detected face regions as the targets of video correction processes.

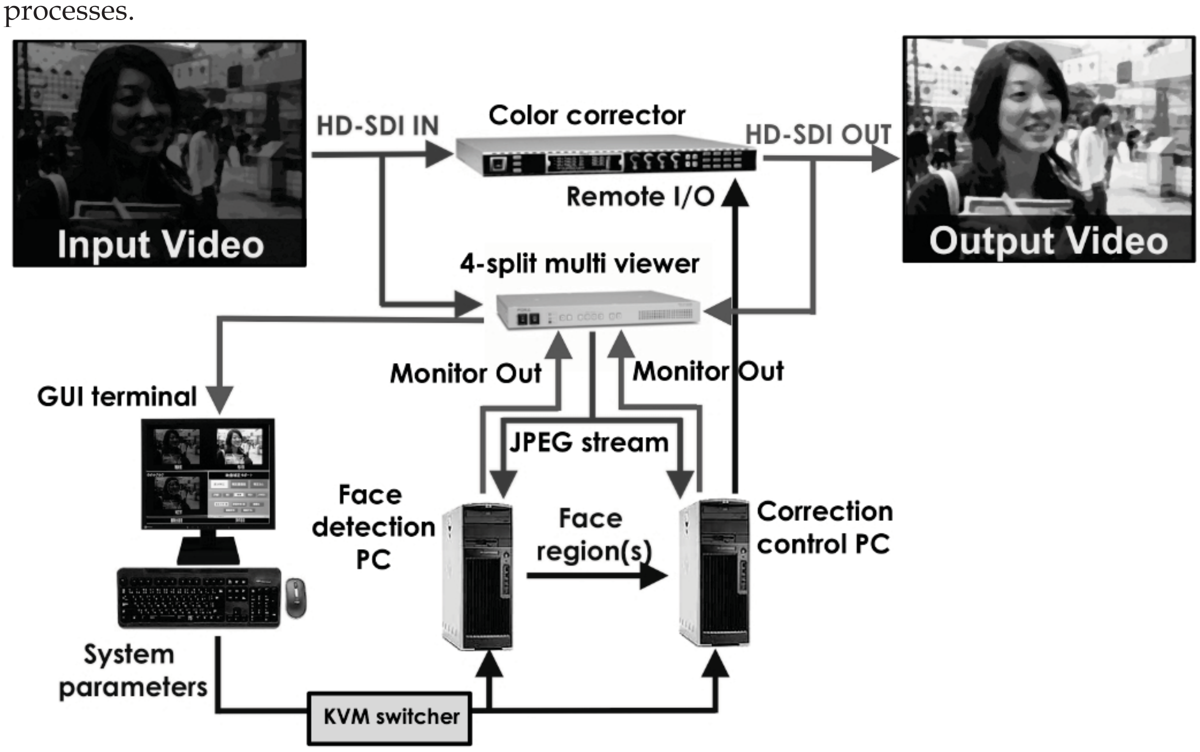

Fig. 6. Overview of the video correction system.

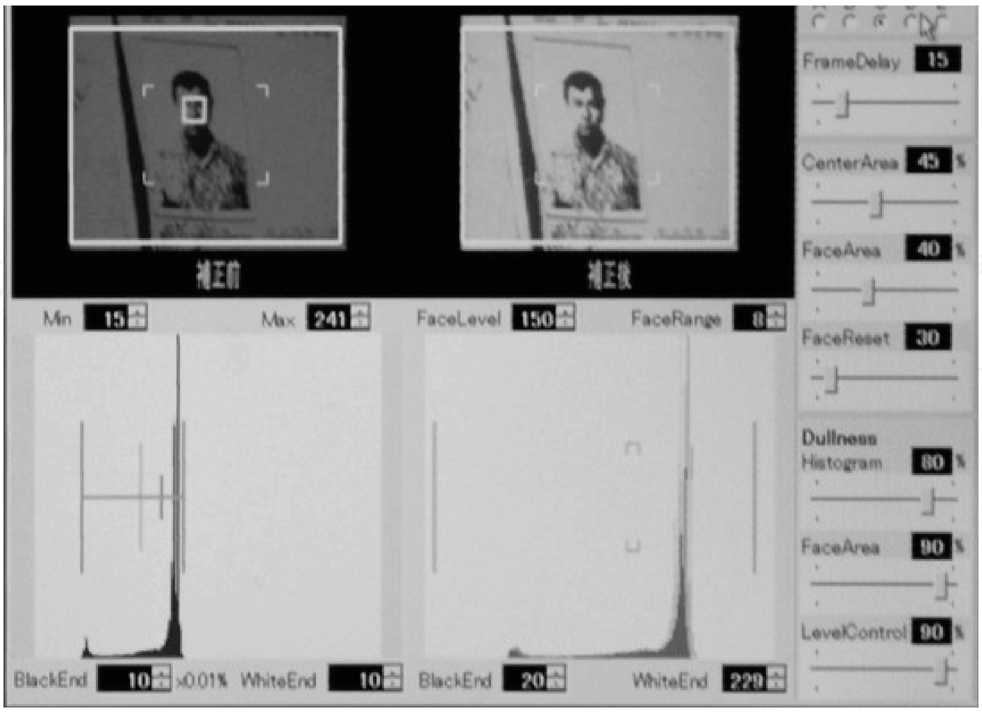

Fig. 7. Example of the GUI screen in parameter adjustment/presetting mode.

3.2 Facial Occlusion Spotting

In many cases, a broadcast image sequence is assembled from several video materials, with various visual effects. In this work, we assume a standard video effect, in which some video materials are superimposed on a base image. The main purpose of this section is to spot occlusions, in which the main broadcast subjects are overlapped completely or partially by the superimposed video materials.

Figure 8 shows an overview of our occlusion spotting system. This system is constructed from two parts: a face detection system and an occlusion detector.

Although there are various different situations where occlusions can arise in broadcast scenes, we focus on human faces as a major object of interest in broadcasting. We adopt the following three criteria for a target scene to be detected:

i. Size of the face region is significant.
ii. Spatial ratio of the overlapped area is significant.
iii. The occlusion spans several contiguous frames.

The first criterion reflects a general trend in broadcasting that main image objects are bigger than other incidental objects. The second criterion relates to the degree of occlusion. It reflects a natural assumption that an occlusion is troublesome if it hides more than a certain proportion of the whole face region. The third criterion is for temporal consistency. It aims to exclude accidental occlusions and noise due to camera work and/or movement of the face itself. We show a particular application for a broadcast news program, detecting face occlusions using our face detection technologies and an occlusion detector based on the three criteria described above.

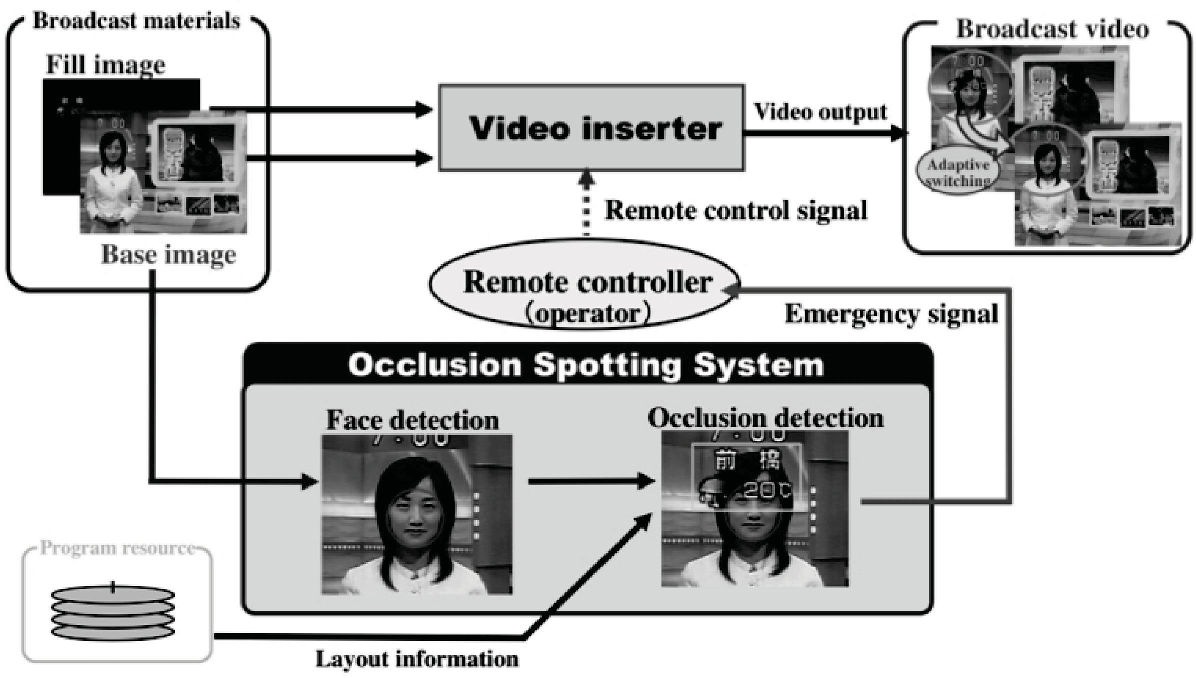

Fig. 8. Architecture of the facial occlusion spotting system.

Occlusion Spotting

Assume a set of face regions $\{ \Omega_i \,;\, i \in I \}$ in an input image D_t at frame t, i.e. regions where the output of a face detector is +1.

According to criteria (i) and (ii), we define the degree of occlusion (occlusion score) for input image D_t as follows:

$$V_t := \max_{i \in I} \left[\lambda_a \phi\left(\frac{r_i}{r_0} ; \rho_a \right) + \lambda_b \phi\left(O_i ; \rho_b \right) \right] \tag{15}$$

$$\phi(x ; \rho) = \begin{cases} 0 & \text{if } x \le 0 \\ x/\rho & \text{if } 0 < x < \rho \\ 1 & \text{otherwise} \end{cases} \tag{16}$$

where $\phi(x; \rho)$ is a score function parameterized by ρ, and the weighting parameters λ_a and λ_b satisfy the following equation:

$$\lambda_a + \lambda_b = 1 \tag{17}$$

Here r_0 is a standard radius for face regions in the given input images. O_i stands for the degree of spatial overlap between the i^{th} face region Ω_i and the target area $\Omega_0 = \{(x, y) : x_R < x < x_L, y_T < y < y_B\}$ to be superimposed. The first term of equation (14) embodies criterion (i), and the second term criterion (ii). As shown in equation (15), we assume that the occlusion score would saturate if the size of face region were large enough, and the overlapping factor were large enough.

We define an importance factor for a pixel (x, y) in the i^{th} face region as a Gaussian distribution, $G(x,y;\ \Omega_i)$. Then we define an overlapping factor O_i as the integral of the distribution $G(x,y;\ \Omega_i)$ inside the target area $0 < O_i < 1$ (see figure 9).

Smoothing of Occlusion Score

From criterion (iii), we apply a smoothing filtering process to the trajectories of occlusion scores, V_t . As a criterion for occlusion, we define the alarm level at time t, U_t . Observing the general trend of temporal occlusions due to pan/tilt movement of the camera and isolated false positive errors in the face detector, we define using a Median filter with buffer length τ .

$$U_t := \text{MED}\ \left[V_{t-\tau+1},\ldots,V_t\right] \tag{18}$$

where $0 < U_t < 1$ since V_t is bounded by [0, 1]. We set $V_t = 0$ ($t < 0$), $t = 2n+1$ ($n \in Z+$). The system assumes there is a continuous occlusion when U_t exceeds a threshold , and outputs warning signals. Therefore, the system response U_t is delayed relative to V_t by $(\tau\text{-}1)/2$. A delay of several frames is acceptable because human operators can not respond in less than several hundreds of milliseconds in an occlusion spotting task.

Figure 10 shows trajectories of V_t and U_t for the proposed scheme. We show examples of true positive and true negative scenes in figures 11 and 12, respectively. Some unstable responses at around $570 < t < 590$ on the trajectories of figure 10 were caused by continuous missing of face regions in some critical scenes, such as scenes in which the target person is walking and shifting his/her head pose. These errors are mainly caused by face detection errors due to pose variations in the broadcast video and insufficient robustness of the face detector we used. We expect that the robustness can be improved if we use a variety of face pose data at the training stage of the face detector, and detectors tuned to multiple poses.

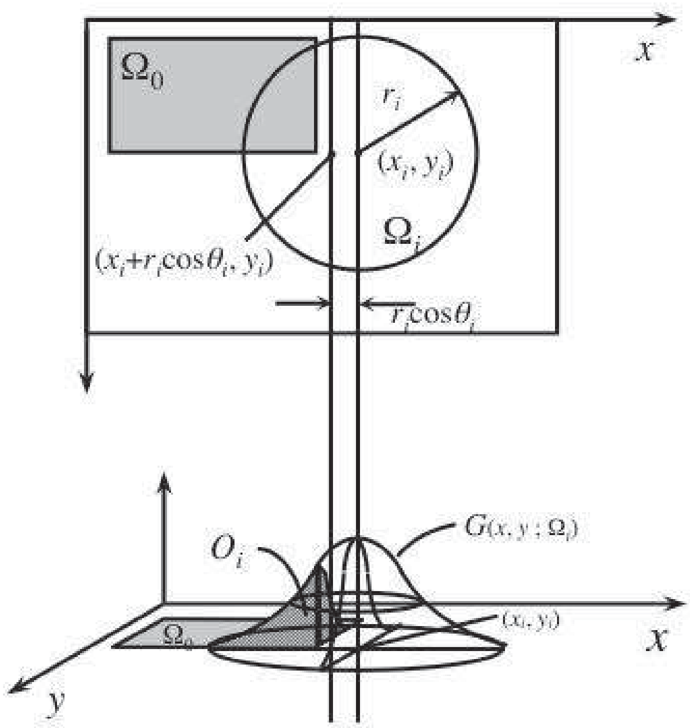

Fig. 9. Importance factor for pixel (x, y) in i^{th} face region.

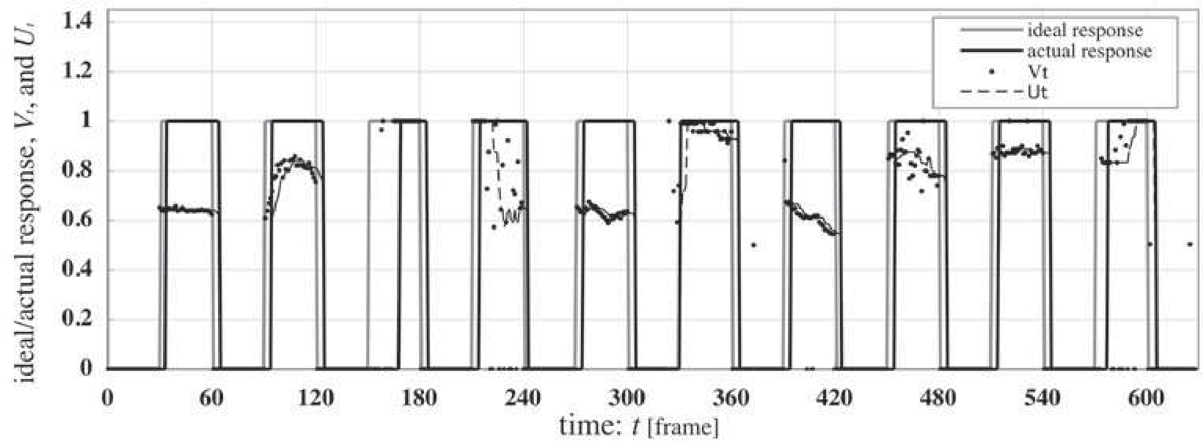

Fig. 10. Trajectories of ideal response, actual response, V_t, and U_t of the system.

Fig. 11. Examples of true positives.

Fig. 12. Examples of true negatives.

4. Conclusion

In this paper we introduced a Bayesian online learning approach, Sequential Importance Search, for face detection in video which dramatically boosts detection speed for sequential input images. The online learning approach successfully prunes a considerable amount of

the search space of face region parameters, and can automatically reset its prediction process in response to discontinuities (cuts) in the input video stream. Experimental results showed the efficiency of the proposed method in comparison to the conventional exhaustive search algorithm, and demonstrated its automatic re-initialization process, Bayesian change detection, at each scene boundary in the input sequence of a simulated broadcasting program compiled from the TRECVID video data. The re-initialization process rests on the sequential model marginal likelihood from the online learning process, which can be derived naturally from the methodology of hierarchical Bayesian inference.
We also introduced two possible applications of our new face detection algorithm, in a video correcting system, and an occlusion spotting system. Although there still remain open issues, these technologies offer the prospect of improved performance in various practical engineering tasks in broadcasting.

5. References

Avidan, S. (2001). Support Vector Tracking, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2001), Vol.1, pp.184-191, 2001.

Doucet, A. (1998). On Sequential Simulation-Based Methods for Bayesian Filtering, CUED/F-INFENG/TR-310, University of Cambridge, 1998.

Freund Y. & Schapire, R. E. (1997). A Decision-theoretic Generalization of On-line Learning and an Application to Boosting, Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119-139, 1997.

Lienhart R. & Maydt J. (2002). An Extened Set of Haar-like Features for Rapid Object Detection", Proceedings of ICIP2002, vol. 1, pp. 900-903, 2002.

Matsumoto T. & Yosui, K. (2007). Adaptation and Change Detection with a Sequential Monte Carlo Scheme, IEEE Transaction on Systems, Man, and Cybernetics, vol. 37, no. 3, pp. 592-606, 2007.

Okuma, K.; Taleghani, A.; Freitas, N. de.; Little, J. J. & Lowe, D. G. (2004) A Boosted Particle Filter: Multitarget Detection and Tracking, Proceedings of ECCV2004, LNCS 3021, pp. 28-39, 2004.

Papageorgiou, C.; Oren, M. & Poggio, T. (1998). A General Framework for Object Detection, Proceedings of ICCV '98, pp. 555-562, 1998.

Platt J. (1999). Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods, Advances in Large Margin Classifiers, MIT Press, 1999.

Viola, P. & Jones, M. (2001). Robust Object Detection using a Boosted Cascade of Simple Features, Proceedings of CVPR2001, vol. 1, pp. 511-518, 2001.

Pattern Recognition

Edited by Peng-Yeng Yin

ISBN 978-953-307-014-8
Hard cover, 568 pages
Publisher InTech
Published online 01, October, 2009
Published in print edition October, 2009

For more than 40 years, pattern recognition approaches are continuingly improving and have been used in an increasing number of areas with great success. This book discloses recent advances and new ideas in approaches and applications for pattern recognition. The 30 chapters selected in this book cover the major topics in pattern recognition. These chapters propose state-of-the-art approaches and cutting-edge research results. I could not thank enough to the contributions of the authors. This book would not have been possible without their support.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821