Improvement of Environmental Adaptivity of Defect Detector for Hammering Test Using Boosting Algorithm

Hiromitsu Fujii[1], Atsushi Yamashita[1] and Hajime Asama[1]

***Abstract*— An automated diagnosis methodology is necessary for the maintenance of superannuated social infrastructures. In this context, the hammering test is an efficient inspection method, and it has been widely used because of the resulting accuracy and efficiency of operation. While robotic automation of the hammering inspection method is highly desirable, the development of an automatic diagnostic algorithm that can operate at actual inspection sites is essential. Furthermore, portability of the diagnostic algorithm is also highly desirable. In this study, in order to construct reliable detectors and to improve their portability for the performance of the hammering test, we propose a boosting-based defect detector that is robust against variations in environmental conditions. In particular, we present the construction of a noise-robust classifier with a refinement of the feature values extracted from hammering sounds and an updating rule of template vectors of its evaluation function. Our experimental results in a concrete tunnel demonstrate the effectiveness of the proposed method; the accuracy of the classifier at an actual site and adaptivity to environmental noise are confirmed.**

I. INTRODUCTION

In recent years, the superannuation of social infrastructure has become a major social problem involving installations such as concrete tunnels and bridges. Particularly in Japan, infrastructures constructed during the era of rapid economic growth are over fifty years old, which is considered the interval of their usefulness. Indeed, some serious accidents have already occurred, and the early detection of problems by continuous inspection of such aging infrastructure has now become indispensable. However, the number of infrastructures that need inspection raises a huge amount [1]. Thus, it is extremely difficult to inspect all of them manually. Moreover, the locations to be inspected, such as those in high and narrow places, are dangerous for workers in many cases. Therefore, the development of an automated inspection system that can be installed on robots such as Fig. 1 is strongly desired.

The hammering test is a kind of inspection method to evaluate infrastructures, and this method uses special sticks called percussion sticks or inspection hammers (Fig. 1). The method has been used for a long time, and it is still widely used for inspection work because of the accuracy of the resulting diagnosis and ease of execution. From the aspect of these advantages, robotic automation of hammering inspection has attracted considerable attention [2], [3], [4].

[1]H. Fujii, A. Yamashita and H. Asama are with the Department of Precision Engineering, Faculty of Engineering, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan. {fujii, yamashita, asama}@robot.t.u-tokyo.ac.jp

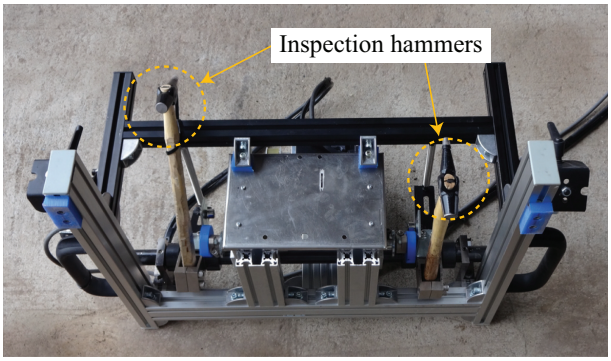


Fig. 1. A prototype of a hammering robot. Such a robot is highly desirable for the inspection of social infrastructures, particularly concrete tunnels.

Our research group has also developed an automated inspection method to detect material defects using the hammering test [5]. Our defect detector is based on a template matching and boosting algorithm [6], which is a machine learning technique. With this approach, it has confirmed that material defects in infrastructures can be detected.

However, the application of this method in environments such as concrete tunnels is problematic. In field conditions, portability of the defect detector becomes a necessity. For example, a detector that is configured for one tunnel should ideally be available to tackle a variety of different situations (i.e., at different times or at other tunnels). In such a case, the difference in the prior learning environment and the actual measurement environment ought to be considered, such as the site's geographical conditions and environmental noises caused by wind and traffic. Building a new detector from scratch for each inspection target is unfeasible and impractical. Thus, a method to calibrate the defect detector to accommodate changes in the working environment is necessary in order to automate inspection at various sites. This issue has not formed the focus of most previous studies on hammer testing.

In the field of transfer learning, many studies on the additional learning of boosting algorithms have been proposed [7], [8], [9]. For example, Pang *et al.* proposed a method that can transfer features of interest and relearn a prior detector by using a weighting loss function with covariate values between the auxiliary training set and target training set. The method uses only a small number of additional training samples to update the prior learner generated by a large number of auxiliary training sets. Many applications of the methods focus on the problem of applying the prior detector

to the real world, since the prior detector is obtained under well conditioned computer graphic (CG) environments. These methods cannot be applied to actual inspection sites, which forms our focus.

Therefore, in this study in order to improve the portability of defect detectors for hammering tests, we propose a boosting based defect detector that is robust to changes in environmental conditions. In particular, we present the construction of a noise-robust classifier with refinement of feature values extracted from hammering sounds and an updating rule of template vector. Our method can be applied not only to problems similar to the prior one, but also to other problems under different environments.

II. NOISE-ROBUST DEFECT DETECTOR

A. Short-Time Fourier Transform

The Fourier transform has been commonly applied to signal processing to obtain the frequency domain representation of a time series. However, for material diagnoses that solve some kind of matching problem, the difficulty arising in performing a frequency analysis is that the shapes of frequency distribution alter according to the sampling timing. Thus, a slight degree of time lag of sampling causes an error of diagnosis. The short-time Fourier transform (STFT) is a method for time-frequency transformation. The method has been used for machine fault detection or nondestructive inspection of infrastructures [10], [11]. In this approach, signals are multiplied by a window function sliding along the time axis and transformed by Fourier transform. The resulting signal in the frequency domain can be analyzed via a time-frequency dimensional representation.

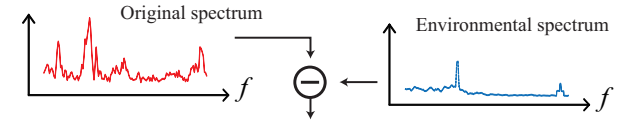
$$\mathrm{STFT}_{x,w}[n,\omega] = \sum_{m=-\infty}^{\infty} x[n+m]\, w[m] e^{-i\omega m} , \qquad (1)$$

where $w[n]$ represents a window function that is a Hanning window function in our study and ω denotes the angular frequency.

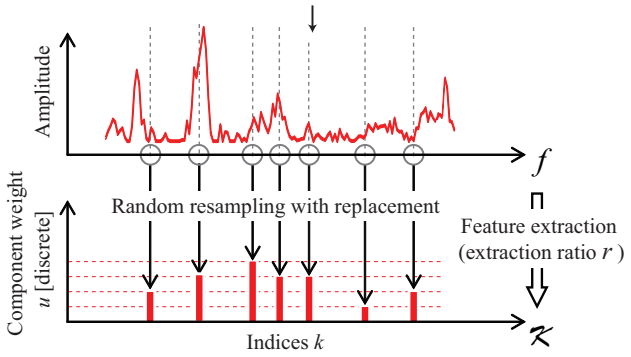
The resulting signals include the frequency signals of any sampling timing. These signals can be considered as a set of feature values of hammering sounds that represent the conditions of an inspection target. In this study, a large number of resulting signals in the frequency domain are used for the learning of a defect detector as training samples. Upon exploiting these signals for learning, the accuracy of diagnosis increases because the single FFT in any timing can be applied to the diagnosis. Furthermore, because a single FFT is fast, our method can be used for online diagnosis.

B. Templatization Using Weighted ZNCC

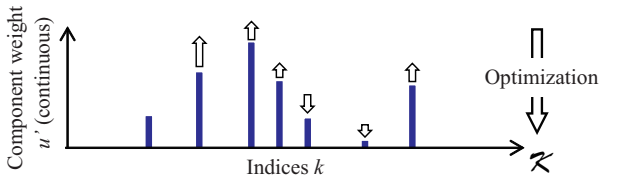
The amplitude scale of the spectrum amplitude can vary according to the impact strength when an inspection target is struck or hit in the hammering test. In order to classify signals robustly without the influence of sound pressure, similarities among signals are evaluated by template matching based on the zero-mean normalized cross-correlation (ZNCC) that is extended by introducing the weights of the



(a) Spectrum subtraction: Spectra of hammering sounds are preprocessed by subtracting environmental noise.



(b) Bootstrapping feature extraction: the components of selected feature vector $\boldsymbol{u}$ are discrete values.



(c) Optimization via the downhill simplex method: the components of the optimized feature vector $\boldsymbol{u}'$ are continuous values.

Fig. 2. Feature refinement of proposed method: Feature extraction using a spectrum subtraction (II-C.1) and bootstrapping (II-C.2) and optimization of feature vector using the downhill simplex method (II-C.3)

frequency components. The evaluation function $S(\boldsymbol{A},\boldsymbol{u},\boldsymbol{x})$ is represented as follows:

$$S(\boldsymbol{A},\boldsymbol{u},\boldsymbol{x}) = \frac{\sum_{k\in\mathcal{K}} u_{(k)}(A_{(k)}-\bar{A})(x_{(k)}-\bar{x})}{\sqrt{\sum_{k\in\mathcal{K}} u_{(k)}(A_{(k)}-\bar{A})^2}\sqrt{\sum_{k\in\mathcal{K}} u_{(k)}(x_{(k)}-\bar{x})^2}} , \qquad (2)$$

where $\boldsymbol{A}$ represents the template vector calculated with the frequency data of the training set in learning step t. Let $\bar{A}$ and $\bar{x}$ denote the average values of $\boldsymbol{A}$ and $\boldsymbol{x}$, respectively. The set $\mathcal{K}$, the details of which are described later, represents the index set of the frequency components used in the classification of $\boldsymbol{x}$. The index k denotes each frequency index in $\mathcal{K}$. The vector $\boldsymbol{u}$ consists of the weights of the frequency components included in $\mathcal{K}$, which represents the feature vector of a hammering sound. The function evaluates the similarity between the template vector $\boldsymbol{A}$ and input vector $\boldsymbol{x}$. The range of $S(\boldsymbol{A},\boldsymbol{u},\boldsymbol{x})$ is $[-1,1]$ and the same as that of the general ZNCC.

Using $S(\boldsymbol{A},\boldsymbol{u},\boldsymbol{x})$, we obtain the classification of $\boldsymbol{x}$ by each weak learner $h(\boldsymbol{x}) \in \{-1,0,1\}$ as

$$h(\boldsymbol{x}) = \mathrm{sign}\left[\left(S({}^{D}\boldsymbol{A},{}^{D}\boldsymbol{u},\boldsymbol{x}) - S({}^{C}\boldsymbol{A},{}^{C}\boldsymbol{u},\boldsymbol{x})\right)/\theta - 1\right], \qquad (3)$$

where θ represents the positive threshold for the classification to be designed. Further, ${}^{D}\boldsymbol{A}$ represents the template vector

created by the training set obtained from defect materials and ${}^{c}\boldsymbol{A}$ the one from "clean" (defect-free) materials. Both the template vectors are calculated considering the weights $w^{(i)}$ of training samples $\boldsymbol{x}^{(i)}$. They are defined as ${}^{D}\boldsymbol{A}(k) = \sum_{i\in\mathcal{I}} w^{(i)}x^{(i)}(k)$, ${}^{c}\boldsymbol{A}(k) = \sum_{j\in\mathcal{J}} w^{(j)}x^{(j)}(k)$, where $\mathcal{I}$ denotes the defect class that is the set of indices of the training set obtained from defect materials, and $\mathcal{J}$ is the clean class. That is to say, $\boldsymbol{x}^{(i)}$ belongs to the defect class if $i \in \mathcal{I}$, and $\boldsymbol{x}^{(j)}$ belongs to the clean class if $j \in \mathcal{J}$. Similarly, the vectors ${}^{D}\boldsymbol{u}$ and ${}^{c}\boldsymbol{u}$ represent the weights of frequency components.

In eqs. (2) and (3), $\mathcal{K}$ and θ represent the parameters that must be designed for the defect detector. In particular, $\mathcal{K}$ represents the feature vector space in which $\boldsymbol{x}$ is compared with the template vector. The parameter $\mathcal{K}$ is optimized simultaneously during defect detector learning. By the optimization of $\mathcal{K}$, an adaptive detector can be constructed to solve the diagnostic problem whose acoustic features can vary according to the soundness or kind of material. Variable θ denotes the threshold that determines the result of classification by the balance between the ZNCC function values of $S({}^{D}\boldsymbol{A}, {}^{D}\boldsymbol{u}, \boldsymbol{x})$ and $S({}^{c}\boldsymbol{A}, {}^{c}\boldsymbol{u}, \boldsymbol{x})$. Further, $\mathcal{K}$ and θ for the defect detector are selected so that the error ratio ϵ is minimized. Parameter ϵ is evaluated repeatedly in trials wherein the plural candidates of the defect detector correspondingly classify all training sets. The selection of $\mathcal{K}$ is described in detail in section II-C.2.

Furthermore, $w^{(i)}$ must be set for each training sample $\boldsymbol{x}^{(i)}$. The method to weight the training samples is described in section III-A.

C. Feature Refinement of Hammering Sound

1) Spectrum Subtraction: In order to improve environmental adaptivity, it is necessary to reduce the influence of changes in environmental sounds, such as wind noise inside tunnels. For this purpose, the spectra of hammering sounds are preprocessed by spectrum subtraction (Fig. 2(a)). Because the background noise varies continuously, the spectrum to subtract is templatized by using sounds obtained before hammering.

2) Feature Extraction using Bootstrapping: The index set $\mathcal{K}$ represents the feature vector space of a hammering sound. In order to obtain an appropriate $\mathcal{K}$ value for the template vector of the detector, the components of $\mathcal{K}$ are selected using *bootstrapping*, which is a random resampling method with replacement (Fig. 2(b)). In the proposed method, let the dimension of the whole feature space be N, and further, we assume that the components are resampled randomly M times with replacement. Therefore, the set of indices of the selected components is the feature vector space $\mathcal{K}$, in which a target sample is compared with the training samples in evaluation. Let $r = M/N$ $(r > 0)$, which is defined as the extraction ratio that is also one of the parameters that must be designed for the detector.

By repeatedly evaluating each result of random resampling, the appropriate template and feature vector to classify the defects of the target material are obtained. In particular, the weights of the components of a feature vector $\boldsymbol{u}$ can be obtained as a result of the replacement in the sampling. However, in this technique, there is the problem that components that hinder accurate detection are present in the feature vector because of the randomness of selection. In order to solve this problem, we combine the techniques of an optimization of the feature vector (described in the next section) and synthesis of learners by boosting (III-A).

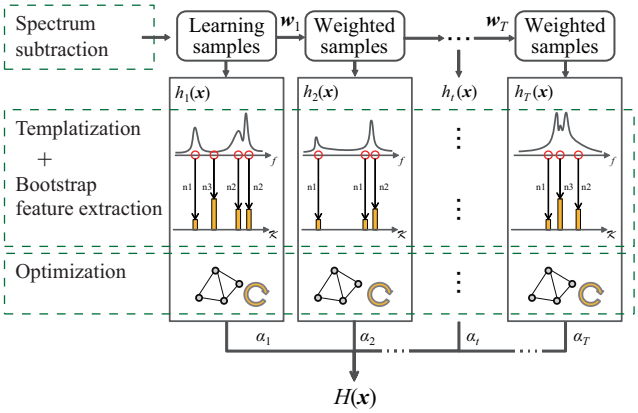


Fig. 3. Schematic of proposed boosting framework. Each weak learner in the proposed boosting framework has two functions: **templatization and extraction** that generate a template vector from training samples by selecting an appropriate frequency component using bootstrapping, and **optimization** that adjusts the weights of the frequency components by a direct search technique.

3) Optimization of Feature Vector: The process of setting the weights of the frequency components that can reduce the error of classification through iterative evaluation is formulated as a nonlinear constrained optimization problem as follows:

$$\begin{aligned} &\text{minimize} : V(\boldsymbol{u}) \\ &\text{subject to} : \boldsymbol{u} \in \mathcal{S}_{\boldsymbol{u}} \ , \end{aligned} \tag{4}$$

where $V(\boldsymbol{u})$ denotes the objective function that is equal to the error ratio ϵ_t described in subsection III-A; $\boldsymbol{u}$ denotes a vector that consists of optimized parameters that are equal to the weights in the frequency domain; and $\mathcal{S}_{\boldsymbol{u}}$ a set of optimized parameters that satisfy certain constraints, which are conditions that must be fulfilled to ensure validity of evaluation.

The difficulty in this optimization problem lies in obtaining the derivatives of the objective function with respect to the optimized parameters. The difficulty arises because the mathematical relation between a sample and the label is not obvious, that is, the error ratio is obtained only through trials. Therefore, we used a direct search technique: the downhill simplex method [12]. The initial values of the optimized vector are significant in this regard. In the proposed method, the initial vector can be given as the result of the bootstrapping selection.

III. UPDATING METHOD

A. Synthesis of Learners by Boosting

This section briefly describes the basis for classifier construction based on boosting [6]. Boosting, an ensemble learning algorithm, is a machine learning meta-algorithm

that can be used to create a strong learner by integrating a set of weak learners. In a boosting algorithm, the whole learner (strong learner) consists of plural learners (weak learners). The strong learner combines the outputs of the weak learners using a weighted majority vote. Each weak learner is created in order along with the updating of the weights of training set. The training set that a previous weak learner misclassified is weighted in the next learning step, so that the weak learner in later steps can focus on the difficult examples that the learners generated in previous steps cannot classify.

The strong learner $H(\boldsymbol{x})$ for a signal $\boldsymbol{x}$ is a binary classifier that can be expressed as

$$H(\boldsymbol{x}) = \mathrm{sign}\left[\sum_{t=1}^{T} \alpha_t h_t(\boldsymbol{x})\right] \in \{-1, 0, 1\} \ , \tag{5}$$

where $h_t(\boldsymbol{x})$ denotes a weak learner in learning step t. Parameter T denotes the number of weak learners that organize a strong learner. Further, α_t denotes the confidence coefficient of each weak learner as computed by the error ratio.

Let $\chi = \left\{(\boldsymbol{x}^{(1)}, y^{(1)}), \cdots, (\boldsymbol{x}^{(I)}, y^{(I)})\right\}$ be labeled training set where I denotes the number of training sets. Let ϵ_t be the error ratio of $h_t(\boldsymbol{x})$, which is calculated by testing all training sets. Subsequently, α_t is calculated as following

$$\alpha_t = \frac{1}{2}\log\left(\frac{1-\epsilon_t}{\epsilon_t}\right) \ , \tag{6}$$

such that the conditions $\epsilon_t \leq 0.5$ and $\alpha_t \geq 0$ are always satisfied in binary classification.

The subscript i denotes the index of the training set, for the i-th set of the training set $\boldsymbol{x}^{(i)}$ and the corresponding their label $y^{(i)}$ $(i = 1, \cdots, I)$. The weights $w_t^{(i)}$ of $\boldsymbol{x}^{(i)}$ are updated at the end of learning step t as

$$w_{t+1}^{(i)} = w_t^{(i)} e^{-\alpha_t y^{(i)} h_t(\boldsymbol{x}^{(i)})} \ , \tag{7}$$

At the beginning of each learning step, $w_t^{(i)}$ is normalized as $\sum_i w_t^{(i)} = 1$. As a result, if the example is classified correctly by $h_t(\boldsymbol{x}^{(i)})$, the weight decreases relatively in the next learning step. However, if it is misclassified, the weight increases relatively.

The schematic of the proposed algorithm is shown in Fig. 3. In our algorithm, each weak learner has the feature refinement functions mentioned in section II.

B. Updation of Frequency Template

Let us consider a labeled auxiliary training set $\tilde{\chi} = \left\{(\tilde{\boldsymbol{x}}^{(1)}, \tilde{y}^{(1)}), \cdots, (\tilde{\boldsymbol{x}}^{(I)}, \tilde{y}^{(I)})\right\}$ with an additional training set $\chi' = \left\{(\boldsymbol{x}'^{(1)}, y'^{(1)}), \cdots, (\boldsymbol{x}'^{(J)}, y'^{(J)})\right\}$. Further, I and J represent the numbers of prior training samples and additional samples, respectively. Accordingly, let $\boldsymbol{x}^{*(i)}$ denote the new training samples in the new training set χ^*, which is defined as

$$\boldsymbol{x}^{*(i)} = \begin{cases} \tilde{\boldsymbol{x}}^{(i)} & (1 \leq i \leq I) \\ \boldsymbol{x}'^{(i-I)} & (I+1 \leq i \leq I+J) \end{cases} \ . \tag{8}$$

Let $\tilde{w}_t^{(i)}$ represent the weights of the i-th prior training sample in $\tilde{\chi}$, and $w_t'^{(i)}$ the weights of the additional training sample in χ'. The ratio between both training sets is defined as $\Sigma_{i=1}^{I} \tilde{w}_t^{(i)} : \Sigma_{j=1}^{J} w_t'^{(j)} = \tilde{\gamma}_t : \gamma_t'$, where $\tilde{\gamma}_t$ and γ_t' represent the constant variables in each learning step t. The weight coefficients $\tilde{\gamma}_t$ and γ_t' are calculated from the weight of the number and the confidence of weak learners, respectively, as follows:

$$\tilde{\gamma}_t = \frac{\Sigma_{j=1}^{J}\Sigma_{t=1}^{T}(\frac{1}{2}\,\tilde{\alpha}_t\,(1 + y'^{(j)}\,\tilde{h}_t\,(\boldsymbol{x}'^{(j)})))}{\Sigma_{j=1}^{J}\Sigma_{t=1}^{T}\,\tilde{\alpha}_t} \ , \tag{9}$$

$$\gamma_t' = \frac{\Sigma_{j=1}^{J}\Sigma_{t=1}^{T}(\frac{1}{2}\alpha_t'(1 + y'^{(j)}h_t'(\boldsymbol{x}'^{(j)})))}{\Sigma_{j=1}^{J}\Sigma_{t=1}^{T}\alpha_t'} \ , \tag{10}$$

where $\tilde{\alpha}_t$ represents the confidence coefficient of a prior learner and α_t' the confidence coefficient of an additional learner, which is calculated in the same manner as in eq. (6) in section III-A. At learning step $t = t_c$, let the confidence coefficients in the forwarding steps α_t' $(t_c \leq t \leq T)$ be defined as

$$\alpha_t' = \begin{cases} \alpha_t' & (1 \leq t < t_c) \\ \tilde{\alpha}_t & (t_c \leq t \leq T) \end{cases} \ . \tag{11}$$

Let $w_t^{*(\ell)}$ $(\ell = 1, \cdots, I+J)$ denote the weights of the new trainings samples in $\left\{\tilde{\chi} \cup \chi'\right\}$ as

$$w_t^{*(\ell)} = \begin{cases} \tilde{w}_t'^{(\ell)} & (1 \leq \ell \leq I) \\ w_t'^{(\ell-I)} & (I+1 \leq \ell \leq I+J) \end{cases} \ , \tag{12}$$

where $\tilde{w}_t'^{(\ell)}$ denotes the updated $\tilde{w}_t^{(\ell)}$ that is multiplied by a constant factor. Upon satisfying $\Sigma_{\ell=1}^{I+J} w_t^{*(\ell)} = 1$ and $\Sigma_{i=1}^{I} \tilde{w}_t^{(i)} : \Sigma_{j=1}^{J} w_t'^{(j)} = \tilde{\gamma}_t : \gamma_t'$ in each learning step t, the new weights of training samples are updated as follows:

$$\tilde{w}_t'^{(i)} \leftarrow \frac{\tilde{\gamma}_t}{(\tilde{\gamma}_t + \gamma_t')\Sigma_{i=1}^{I}\,\tilde{w}_t^{(i)}}\ \tilde{w}_t^{(i)} \ , \tag{13}$$

$$w_t'^{(j)} \leftarrow \frac{\gamma_t'}{(\tilde{\gamma}_t + \gamma_t')\Sigma_{j=1}^{J} w_t'^{(j)}}\ w_t'^{(j)} \ . \tag{14}$$

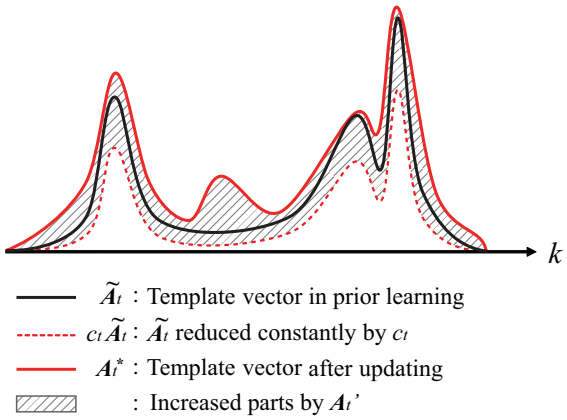


Fig. 4. Schematic of template vector updation in our method as expressed by eq. (17). The original template vector $\tilde{\boldsymbol{A}}_t$ that was obtained in prior learning is reduced constantly by c_t. The reduced vector is supplied by the differential template vector $\boldsymbol{A}_t'$, which is calculated by the weights of additional training samples according to the result of the additional tests.

(a) (b) (c)

Fig. 5. Experimental environment and devices: (a) An example of defect in concrete material (delamination), (b) Measurement of hammering sounds and inspection experiment on high-lift work vehicle, (c) Measurement equipment and prototype of hammering unit

By considering the updating rules of the training sample weights expressed by eqs. (13) and (14), the updating rules of the template vector can be obtained. In a manner similar to the process described in section II-B, the template vector is calculated by considering the weights of trainings samples as follows:

$$A_t^*(k) = \sum_{i=1}^{I+J} w^{*(i)} \boldsymbol{x}^{*(i)} \quad (15)$$

$$= \sum_{i=1}^{I} \tilde{w}_t'^{(i)} \tilde{x}^{(i)}(k) + \sum_{j=1}^{J} w_t'^{(j)} x'^{(j)}(k) \quad (16)$$

$$= c_t \tilde{A}_t(k) + A_t'(k) , \quad (17)$$

where $c_t = \tilde{\gamma}_t / ((\tilde{\gamma}_t + \gamma_t') \Sigma_{i=1}^{I} \tilde{w}_t^{(i)})$, which is the constant appearing in eq. (13). Let $\tilde{A}_t(k) = \sum_{i=1}^{I} \tilde{w}_t^{(i)} \tilde{x}^{(i)}(k)$, which is the k-th component of the auxiliary template vector. Parameter k represents the frequency index in the feature space $\mathcal{K}$. Let $A_t'(k) = \sum_{j=1}^{J} w_t'^{(j)} x'^{(j)}(k)$, which is the k-th component of the template vector from the additional training samples. The vector can be calculated using the weights of the samples, which are updated in the similar manner of eq. (7) as

$$w_{t+1}'^{(j)} = w_t'^{(j)} e^{-\alpha_t' y'^{(j)} h_t'(\boldsymbol{x}'^{(j)})} . \quad (18)$$

Noting, as these updation rules of eqs. (17) and (18) indicate, the whole learner can be updated by only the values that are calculated in prior learning and the additional training set in $\mathcal{X}'$.

The schematic of the updation of the template vector is shown in Fig. 4. In each learning step, the original template vector $\tilde{\boldsymbol{A}}_t$ that was obtained in prior learning is reduced constantly by c_t. The differential template vector $\boldsymbol{A}_t'$ supplies the reduced vector with the weights of the additional training samples according to the result of the additional tests. The new template vector $\boldsymbol{A}_t^*$ is the sum of the reduced vector and the differential vector.

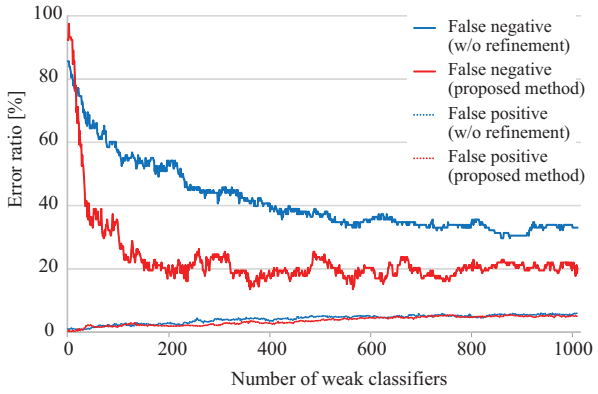


Fig. 6. Feature refinement result of the relation between the number of weak learners and the error ratio of evaluation samples that were not included in training samples. The horizontal axis represents the number of weak learners, and the vertical axis represents the error ratio [%]. Because of the synthesizing multiple learners, the error ratio of false negatives decreased. The best (smallest) error ratio of false negatives as obtained by the proposed detector was 13.6%.

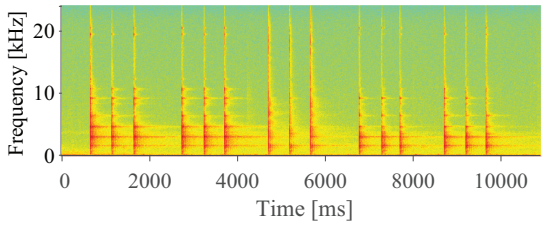
IV. EXPERIMENTS IN CONCRETE TUNNEL

In order to confirm the validity of the proposed method, inspection experiments were conducted in a concrete tunnel.

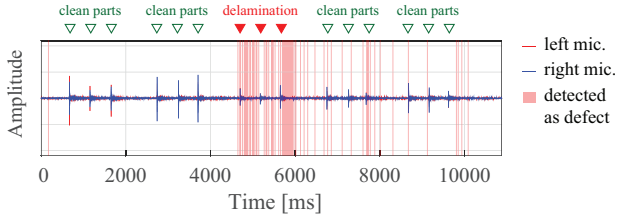
A. Experimental Settings

With regards to various kinds of concrete defects, delamination is considered to be a particularly serious defect because pieces that delaminate from a wall frequently fall down and damage cars and injure pedestrians. In the experiments, we thus particularly focused on the detection of delamination. An example of delamination is shown in Fig. 5(a). While the detection of delamination is highly desirable, the process is complex. This is because delamination is a complex phenomenon that can arise, for example, due to closing cracks or expansions of rusted reinforcing iron.

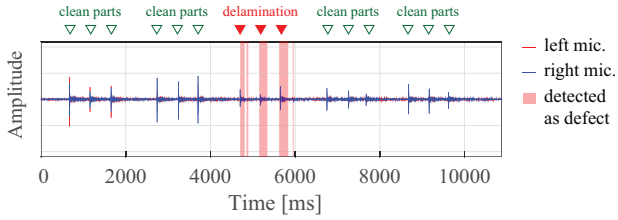
Hammering sounds were recorded at a concrete tunnel in Kanagawa Prefecture in Japan, wherein a road was closed temporarily for the experiments. In order to obtain the



(a) An example of spectrogram of hammering work



(b) Result of defect detection by the detector without feature refinement



(c) Result of defect detection by the detector with feature refinement

Fig. 7. Comparison of the detection results between the prior detector (Fig. 7(b)) and the updated detector (Fig. 7(c)): Hammering motions were executed 15 times in 10 seconds, which can be confirmed in both figures as signal peaks. They consisted of three strikes against defect parts of delamination (between 4,500 ms and 6,000 ms), and the other twelve strikes against clean parts. The areas emphasized by the half-tone background indicate the time intervals corresponding to the detection of defects.

signals classified between cleans and defects, the tunnel was examined by professional inspection workers in advance. The inspection works and measurements were conducted using a high-lift work vehicle (Fig. 5(b)).

The experimental equipment used in our study is shown in Fig. 5(c). Two condenser microphones and a recorder were used for recording hammering sounds. The resolution and sampling rate were 24 bit and 48.0 kHz respectively. As diagnostic tools, we used inspection hammers, which are generally used for inspection of concrete infrastructures. The diameter and the weight of the hammer head were 12.4 mm and 0.1 kg, respectively.

B. Experimental Results

1) Result of Feature Refinement: Figure 6 shows the effectiveness of feature refinement described in section II-C. The horizontal axis represents the number of synthesized weak learners, and the vertical axis represents the error ratio [%] of detection. The red lines indicate the results of the proposed detector that consist of the weak learners constructed with the feature refinement, and the blue ones indicate the results of the detector with only bootstrapping feature extraction (II-C.2). The solid lines denote the results of false negatives that misclassified defects as cleans, and the dotted lines denote the results of false positives that misclassified cleans as defects. These values were evaluated by obtaining the average of five-fold cross validation using the training set obtained under different situations (i.e., time and place), which training set consists of 6,926 sound samples including 1,010 delamination samples and 5,916 clean samples.

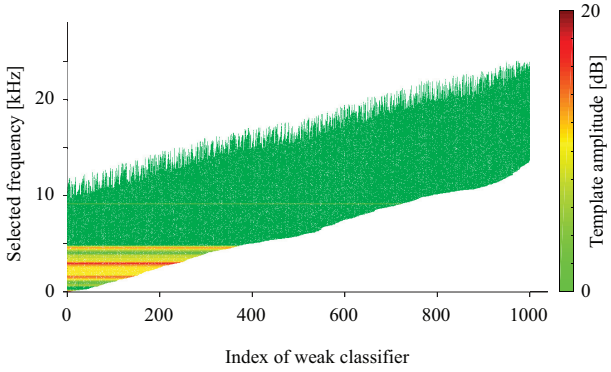
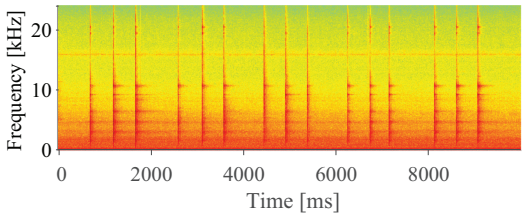


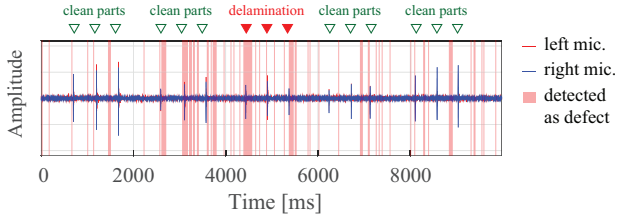
Fig. 8. Frequency components selected by each weak learner. The whole detector (strong learner) consisted of 1,024 weak learners. With each learner focusing on different frequency bands, the accuracy of strong learner was enhanced.

We were able to confirm that the feature refinement is effective to detect delamination accurately. As a rough trend, as the number of weak classifiers increased, the error ratio of false negatives decreased. In contrast, the error ratio of false positives deteriorated slightly as the number of weak classifiers increased. However, the degree of deterioration was quite small. Moreover, reduction of false negative case is a greater concern in this kind of inspection application.

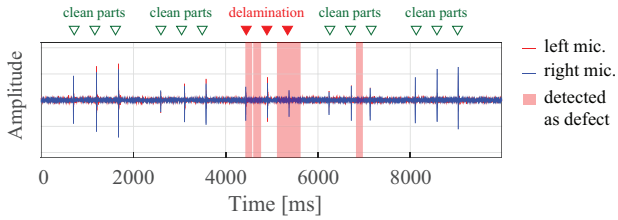
The result of delamination detection is shown in Fig. 7. In the experiment, a professional inspection worker struck the concrete wall 15 times continuously in 10 seconds. The fifteen trials consisted of twelve hammer strikes against clean parts (strikes 1st to 6th and 10th to 15th), and three hammerings against delamination parts (strikes 7th to 9th). Figure 7(a) shows the spectrogram measured during hammering, the horizontal axis represents time [ms], the vertical axis represents frequency [kHz], the depth of color indicates the amplitude strength in the frequency domain. The moments of the impacts can be confirmed from sharp changes of the spectrogram color. The comparison of the time-domain detection results between the detector without feature refinement and the proposed method is shown in Figs. 7(b) and 7(c), respectively. In both figures, the horizontal axis represents time [ms], the vertical axis represents the normalized signal amplitude, and the areas emphasized by the half-tone background indicate the time intervals corresponding to the detection of the defects by each detector. From these results, we observe that the proposed detector with feature refinement can detect delamination more accurately than the one without feature refinement. For example, by applying the proposed detector, the hammering sounds corresponding to delamination between 4,500 and 6,000 ms were detected



(a) An example of spectrogram of hammering with engine noise



(b) Result of delamination detection without updation with engine noise; The loud noise hinders accurate detection.



(c) Result of delamination detection by the updated detector with engine noise; Because of updation, the accuracy of detection was maintained.

Fig. 9. Comparison of the detection results with the presence of engine noise between the prior detector without updation (Fig. 9(b)) and the updated detector (Fig. 9(c))

more distinctly, and the other clean signals can be detected as cleans correctly.

The process of how the weak classifiers of the detector were constructed is shown in Fig. 8. The horizontal axis represents the index t of the weak classifiers h_t, and the vertical axis represents the frequency band [kHz], wherein each classifier evaluates signals. The depth of color denotes the amplitude of the template vector. Here, we remark that the weak classifiers are sorted in order of their lowest limits of frequency bands to improve the viewing clarity of the figure. The whole detector consisted of multiple weak learners that focused on various frequency bands. It can be understood that the diversity of learners contributed to the improvement of the accuracy of detection.

2) Updation Against Environmental Noise: Furthermore, in order to verify the effectiveness of the proposed method against environmental noise, experiments were conducted with the engine of the high-lift work vehicle kept running. The detector was updated by the updation method described in section III using the acoustic samples obtained with the engine noise. For additional learning, 2,690 training sets were used, which included the engine noise and hammering sounds together.

An example of the spectrogram of the hammering sound

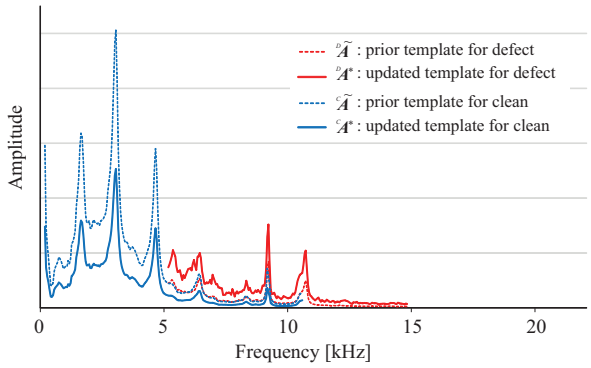


Fig. 10. Result of template updation: The new template vector was updated based on the prior template vector that was reduced constantly by a factor of c_t in eq. (17). Although $\boldsymbol{A}^{*}$ basically retained the shape of the prior template vector according to eq. (17), the shape changed partly.

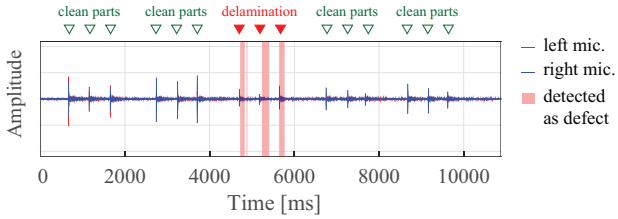


Fig. 11. Detection result for signals shown in Fig. 7(b) and Fig. 7(c) with the use of the updated detector. This result indicates that the updated detector maintains the diagnostic accuracy against the kind of defects of prior learning.

with the engine noise is shown in Fig. 9(a), similar to Fig. 7(a). In this case, the signal-to-noise ratio of the engine noise was around 27 dB. A comparison of the detection results between the prior detector and the updated detector is shown in Figs. 9(b) and 9(c). Similar to the experiment for verifying the feature refinement, hammering motions were executed fifteen times in 10 seconds. Both figures follow the same plot format as Figs. 7(b) and 7(c). In particular, from Fig. 7(b), we note that because of the adverse influence of noise, the accuracy of the detection by the prior detector decreases considerably. A comparison of two sets of results confirms that the updated detector is capable of detecting defects accurately.

The result of template updation is shown in Fig. 10. The horizontal axis represents frequency, and the vertical axis represents amplitude. The shape of the updated template vector $\boldsymbol{A}^{*}$ is essentially the same as that of the prior template vector according to eq. (17). However, as a result of updation, the shape changed partly. These frequency components were considered as significant for new detection tasks. Figure 11 shows the detection result for the set of signals illustrated in Figs. 7(b) and 7(c). By retaining the original template shape, the updated detector could address the defects for the prior one.

In summary, we applied our method to construct a highly accurate delamination defect detector using the hammering signals obtained in an actual concrete tunnel, which is an

accurate representation of a real-world scenario. Furthermore, with the application of the proposed updation method, the existing detector could be updated to the one whose performance is robust to environmental noise.

V. CONCLUSION AND FUTURE WORK

In this study, in order to improve portability of defect detectors for hammering tests, we proposed and demonstrated a boosting-based defect detector that is robust to environmental changes. A construction method of a noise-robust classifier with refinement of feature values extracted from hammering sound and an updating rule of the template vector are presented. Our method can be applied not only to problems similar to the prior one, but also to other problems under different environments. Our experimental results showed the proposed method can accurately distinguish signals under conditions of loud environmental noise.

As a future work, we plan to apply the proposed method to tasks under running noises of robotic hammering unit. For accuracy improvement of the detector, the template vectors have to deform in order not to decrease the accuracy of the classification for the prior problem. The evaluation of vector deformation is our future work.

ACKNOWLEDGEMENTS

This work was supported in part by the Cross-ministerial Strategic Innovation Promotion Program (SIP) of the New Energy and Industrial Technology Development Organization (NEDO), Grant-in-Aid for JSPS Fellows 269039, and Institute of Technology, Tokyu Construction Co., Ltd.

REFERENCES

[1] Ryohei Takada, Naoki Oishi, “Priority Issues of Infrastructure Inspection and Maintenance Robot: A part of COCN 2012 project ‘disaster response robot and its operational system’ ”, Humanitarian Technology Conference (R10-HTC), 2013 IEEE Region 10, pp. 166–171, 2013.

[2] Takeshi Suda, Atsushi Tabata, Jun Kawakami, Takatsugu Suzuki, “Development of an Impact Sound Diagnosis System for Tunnel Concrete Lining”, Tunneling and Underground Space Technology, Vol. 19, Issue 4–5, pp. 328–329, 2004.

[3] Atsushi Yamashita, Takahiro Hara, Toru Kaneko, “Inspection of Visible and Invisible Features of Objects with Image and Sound Signal Processing”, Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3837–3842, 2006.

[4] Fumihiro Inoue, Satoru Doi, Tatsuya Ishizaki, Yasuhiro Ikeda, Yutaka Ohta, “Study on Automated Inspection Robot and Quantitative Detection of Outer Tile Wall Exfoliation by Wavelet Analysis”, Proceedings of International Conference on Control, Automation and Systems 2010, pp. 994–999, 2010.

[5] Hiromitsu Fujii, Atsushi Yamashita, Hajime Asama, “Automated Diagnosis of Material Condition in Hammering Test Using a Boosting Algorithm”, Proceedings of the 2014 IEEE Workshop on Advanced Robotics and its Social Impacts, pp. 101–107, 2014.

[6] Yoav Freund and Robert E. Schapire, “A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting”, Journal of Computer and System Sciences, Vol. 55, Issue 1, pp. 119-139, 1997.

[7] Nikunj C. Oza, “Online Bagging and Boosting”, International Conference on Systems, Man and Cybernetics, Vol. 3, pp. 2340–2345, 2005.

[8] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, Yong Yu, “Boosting for Transfer Learning”, In Proceedings of the International Conference on Machine Learning, pp. 193-200, 2007.

[9] Junbiao Pang, Qingming Huang, Shuicheng Yan, Shuqiang Jiang, Lei Qin, “Transferring Boosted Detectors Towards Viewpoint and Scene Adaptiveness”, IEEE Transactions on Image Processing, Vol. 20, No. 5, pp. 1388–1400, 2011.

[10] Mustapha Mjit, Pierre-Philippe J. Beaujean, David J. Vendittis, “Comparison of Fault Detection Techniques for an Ocean Turbine”, Annual Conference of the Prognostics and Health Management Society, pp. 123–133, 2011.

[11] Marco Cocconcelli, Radoslaw Zimroz, Riccardo Rubini, Walter Bartelmus, “STFT Based Approach for Ball Bearing Fault Detection in A Varying Speed Motor”, Condition Monitoring of Machinery in Non-Stationary Operations, Springer Berlin Heidelberg, pp. 41–50, 2012.

[12] John A. Nelder, Roger Mead, “A Simplex Method for Function Minimization”, The Computer Journal, Vol. 7, No. 4, pp. 308-313, 1965.