# Integrated Image Sensor and Hyperparameter Optimization of Convolutional Neural Network for Facial Skin Detection

Hsueh-Yi Lin,[1] Cheng-Jian Lin,[1,2*] Shiou-Yun Jeng,[3] and Cheng-Yi Yu[1]

[1]Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan
[2]College of Intelligence, National Taichung University of Science and Technology, Taichung 404, Taiwan
[3]Department of Business Administration, Asia University, Taichung 413, Taiwan

Detection of the facial skin condition enables people to better understand skin problems and helps them select appropriate treatment methods and maintenance products. We collected facial images of different skin problems by using an image optical sensor. To overcome the problem of insufficient training data, the collected dataset was expanded through data augmentation. In the field of computer vision, deep learning is often used for solving image recognition problems with high accuracy. Therefore, we used a convolutional neural network (CNN) to detect facial images of different skin problems. To overcome the difficulty of parameter selection and increase the recognition rate and robustness of the CNN, the parameters of the CNN were optimized using the Taguchi method. Eight control factors in the convolutional layer and the L36 orthogonal array (OA) were used in experiments. Analysis of variance was used for statistical analysis in the design of the experiments to obtain the optimal parameter combination for the developed CNNs. The experimental results indicate that the CNN optimized using the Taguchi method had an accuracy of 86.95%. The accuracy of the optimized CNN was 7.24% higher than that of the original CNN. The experimental results prove that the proposed hyperparameter optimization method can effectively improve the accuracy of network detection.

## 1. Introduction

Artificial-intelligence image detection has begun to receive increasing attention in facial cosmetology because it can be used to detect skin conditions and identify an appropriate treatment strategy.[1,2] The complex spatial structures of the skin and lesions associated with continuous skin color changes, pigmented spots, wrinkles, red skin, and acne can be identified; however, such identification is limited by the poor discrimination ability of the human eye. To assist skin analysis experts in dermatology and cosmetology, an image processing system must be able to identify the aforementioned skin conditions.[3] A suitable automatic system for facial condition detection must be able to detect facial conditions from images. Certain challenges,

---

*Corresponding author: e-mail: cjlin@ncut.edu.tw

such as a lack of training data (deep learning requires a large dataset), exist in facial image detection based on deep learning.[4,5] To address these challenges, in this study, we used various data augmentation methods to increase the size of the training set, and deep learning and parameter optimization were conducted to detect different types of skin problems. For image recognition, deep learning is superior to conventional image processing because it does not require users to determine the image capture characteristics.[6]

Among the deep learning methods, convolutional neural networks (CNNs) have achieved considerable success in image analysis applications, such as object detection, facial recognition, and medical image classification.[7–9] CNNs usually consist of fully connected layers (e.g., input layer and output layer), convolutional layers, and pooling layers for downsampling.[10] CNNs are the most commonly used deep learning architecture, and numerous CNN architectures, such as LeNet-5, AlexNet, and GoogleNet, are available.[11–13] LeNet-5 is superior to traditional image processing methods.[14] Lin *et al*.[15] proposed the LeNet-5 architecture and used it to classify breast histopathology images effectively. In the current study, we applied LeNet-5 in facial image detection. This network requires a large number of training parameters, and its parameter setting is complicated.[16] Therefore, the parameters of LeNet-5 must be optimized. The Taguchi method is commonly used for optimizing the parameters of LeNet-5 to increase its classification accuracy. It is a useful method for improving the design of experiments according to the parameters, system, and tolerance, and is widely used in quality assurance systems for the statistical analysis of collected data.[17] The Taguchi method is also useful for determining the optimal parameter combination with minimum experimentation and the order of importance of control parameters.[18] Moreover, it is a robust approach for optimizing the control parameters.[19] In the Taguchi method, an orthogonal array (OA) is used. This array comprises horizontal level factors and other factors that are mapped during an experiment.[20]

The Taguchi method is used to determine the optimal parameter combination for deep learning networks. The problem of the complicated parameter setting process in CNNs can be overcome by using the Taguchi method, which can obtain the optimum parameters in a relatively short time with a relatively low cost and a small number of experiments.[21] Lin *et al*.[9] used the Taguchi method to determine the optimum parameter combination for a CNN structure to improve the accuracy of lung cancer classification. On the basis of the aforementioned discussion, in this paper, we propose a CNN with Taguchi parametric optimization for facial image detection applications. The main aim of this study was to use the Taguchi method to optimize the parameters of a CNN for facial image detection applications. The contributions of this study are described as follows. First, the optimal combination of parameters, such as the convolution kernel size, number of filters, stride, and padding, for a CNN is obtained by using the Taguchi method. Second, the Taguchi method is used to reduce the number of experiments, and thus the time required for facial image detection, and to increase the accuracy of facial image detection.

The remainder of this paper is organized as follows. Section 2 describes the Taguchi method, adopted materials, and developed CNN model. Section 3 presents the experimental results obtained when using the Taguchi method in the developed CNN. Finally, Sect. 4 presents the conclusions of this study and recommendations for future research.

## 2.    Materials and Methods

We developed a CNN to detect facial skin conditions with high accuracy. We used the Taguchi method to optimize the parameters of the CNN and the analysis of variance (ANOVA) for statistical analysis. ANOVA was used to identify the parameters that affected the accuracy of the developed CNN model, then the optimal parameter combination of the CNN was obtained using the Taguchi method. Figure 1 shows the framework of the CNN developed with the Taguchi optimization used to improve the classification performance. This study involved three main steps. First, the developed CNN model was used to detect normal skin, pigmented spots and wrinkles, and red skin and acne (Sect. 2.1). Next, the Taguchi method was used to determine the optimal parameter combination for the developed CNN architecture (Sect. 2.2). Finally, facial images were collected to test the CNN model (Sect. 2.3).

### 2.1    CNN architecture

We applied the LeNet-5 CNN model for facial image detection. As displayed in Fig. 2, the LeNet-5 architecture comprises four parts: a convolution layer, a pooling layer, a fully connected layer, and an activation function. The LeNet-5 architecture used in this study comprises three convolutional layers, two max pooling layers, and two fully connected layers.
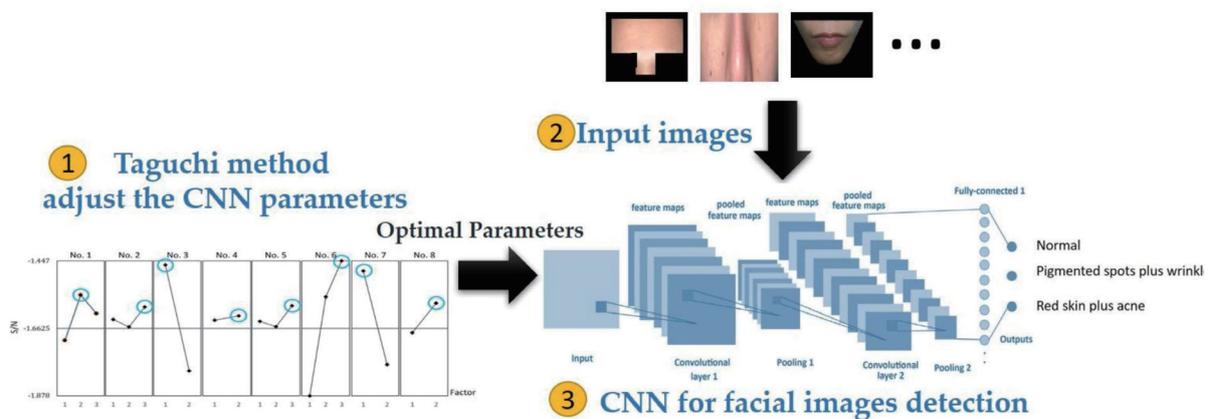


Fig. 1.    (Color online) Framework of the developed CNN with Taguchi optimization.
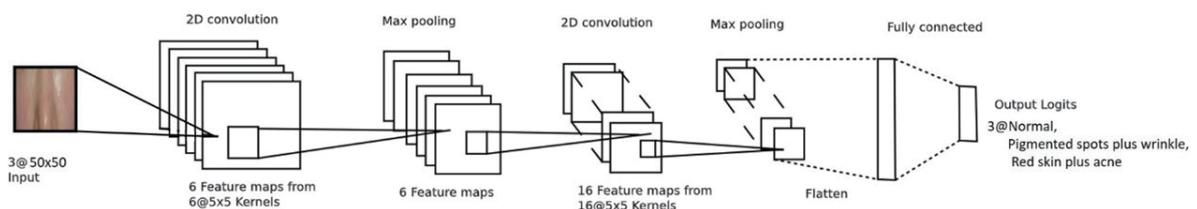


Fig. 2.    (Color online) Basic structure of LeNet-5.

Images with a size of 50 × 50 × 3 pixels are input into the adopted LeNet-5 network structure. The first convolutional layer contains six filters with a convolution kernel size of 5 × 5 pixels. The second convolutional layer contains 16 filters with a convolution kernel size of 5 × 5 pixels. Full zero padding is adopted in both these layers. The third convolutional layer contains 120 filters with feature maps with a size of 1 × 1 pixel. Two max pooling layers are inserted between the three convolutional layers. These layers reduce the number of feature dimensions and maintain max pooling of kernel size and stride as 2 × 2 pixels. To categorize samples as benign or malignant, a fully connected layer is used with the rectified linear unit (ReLU) nonlinear activation function. The kernel, stride, padding, and filter can have different sizes. The parameters of the proposed CNN architecture are presented in Table 1.

We determined the detection performance of the proposed CNN architecture after Taguchi parametric optimization in terms of the accuracy, recall, precision, and false positive rate (FPR), which were derived from a confusion matrix. The diagonal elements of the confusion matrix indicate the number of correctly identified images, and the off-diagonal elements represent the incorrectly identified images. The accuracy, recall, precision, and FPR are expressed as follows:[22]

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN},\tag{1}$$

$$Recall = \frac{TP}{TP + FN},\tag{2}$$

$$Precision = \frac{TP}{TP + FP},\tag{3}$$

$$FPR = \frac{FP}{FP + TN},\tag{4}$$

Table 1
Parameters of the proposed CNN architecture.

| Layer | Image Size | Kernel Size | Stride | Padding | Filter |
|---|---|---|---|---|---|
| Input | 50 × 50 × 3 | — | — | — | — |
| Conv 1 | — | 5 × 5 | 1 | 0 | 6 |
| ReLU | — | — | — | — | — |
| Max pooling 1 | — | 2 × 2 | 2 × 2 | — | — |
| Conv 2 | — | 5 × 5 | 1 | 0 | 16 |
| ReLU | — | — | — | — | — |
| Max pooling 2 | — | 2 × 2 | 2 × 2 | — | — |
| Conv 3 | — | 1 × 1 | 1 | 0 | 120 |
| Fully connected | — | — | — | — | 84 |
| Fully connected | — | — | — | — | 3 |

where TP, FP, TN, and FN indicate the numbers of true positives, false positives, true negatives, and false negatives, respectively. The model included two classes, namely, P and N. Accuracy indicates the rate of correct predictions across the entire dataset; test sensitivity indicates the ability to identify patients with facial skin diseases correctly; and the clinical specificity of a test indicates the extent to which the test correctly identifies patients without the disease.

## 2.2    Taguchi method

In this study, the Taguchi method was used to find the optimal parameter combination for the CNN architecture. The Taguchi method is a design of experiments method, which is a statistical approach that involves using an OA to collect experimental data for analysis. A flowchart of the Taguchi method is illustrated in Fig. 3.

The steps in the Taguchi method are as follows:

**Step 1**: Define the experimental problem

**Step 2**: Determine the quality characteristics

The signal-to-noise (S/N) ratio refers to the ratio between the desired value and undesired factors and was selected as the optimization criterion in this study. To determine the optimal
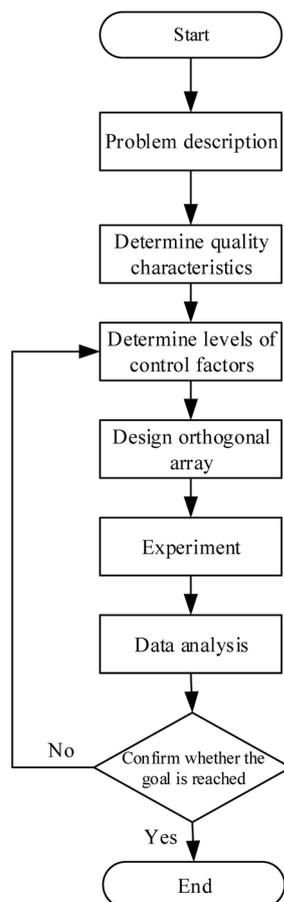


Fig. 3.    Flowchart of the Taguchi method.

process parameters for predicting model performance, the S/N ratio was optimized according to the important factors. The S/N ratio can be analyzed in terms of three performance characteristics: larger-is-better, nominal-is-better, and smaller-is-better characteristics.[23] In this study, the larger-is-better performance characteristic was preferred.

$$\frac{S}{N} = -10\log\left(\frac{1}{n}\sum_{i=1}^{n}\frac{1}{y_i^2}\right),$$

(5)

where $n$ is the number of repetitions in a trial and $y$ is the output of the network.

**Step 3**: Determine the levels of control factors

In this study, eight control factors and a mixture of levels 2 and 3 were adopted. Table 2 presents the control factor parameters of the developed CNN, where A–D and E–H are the parameters of the first and second convolutional layers, respectively.

**Step 4**: Prepare an OA for the experiment

In this study, an OA table was used to collect experimental data for analysis. To reduce the number of experiments and improve the credibility of the results, the L36 orthogonal table presented in Table 3 was adopted.

**Step 5**: Perform the experiment

When the S/N ratio is closer to the target value, the quality loss is lower. The optimal experimental results are obtained by calculating the S/N ratio of each factor and level. We used three observations to calculate the S/N ratio. The observed value was equivalent to the test accuracy under the same conditions in each set of experiments. The higher the average accuracy was in the optimization process, the higher the S/N ratio was. A total of 36 experimental runs were generated using Minitab.

**Step 6**: Conduct data analysis

ANOVA was used to determine the significance of the process parameters.[24] The degree of freedom (DOF), sum of squares (SS), mean of square (MS), F-test significance (F), and percentage contribution (PC) were calculated as follows:

$$DOF = K_A - 1,$$

(6)

Table 2
Control factor parameters of the developed CNN.

| No. | Factor | Level 1 | Level 2 | Level 3 |
|-----|--------|---------|---------|---------|
| A | conv1_Kernel size | 3 | 5 | 7 |
| B | conv1_Filter | 4 | 6 | 12 |
| C | conv1_Stride | 1 | 2 | — |
| D | conv1_Padding | 0 | 1 | — |
| E | conv2_Kernel size | 3 | 5 | 7 |
| F | conv2_Filter | 8 | 16 | 32 |
| G | conv2_Stride | 1 | 2 | — |
| H | conv2_Padding | 0 | 1 | — |

Table 3
OA for experimental parameter setting.

| | Factor | | | | | | | | | Factor | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Run | A | B | C | D | E | F | G | H | Run | A | B | C | D | E | F | G | H |
| 1 | 3 | 4 | 1 | 0 | 3 | 8 | 1 | 0 | 19 | 3 | 4 | 1 | 1 | 7 | 32 | 2 | 0 |
| 2 | 5 | 6 | 1 | 0 | 5 | 16 | 1 | 0 | 20 | 5 | 6 | 1 | 1 | 3 | 8 | 2 | 0 |
| 3 | 7 | 12 | 1 | 0 | 7 | 32 | 1 | 0 | 21 | 7 | 12 | 1 | 1 | 5 | 16 | 2 | 0 |
| 4 | 3 | 4 | 1 | 0 | 3 | 16 | 1 | 0 | 22 | 3 | 6 | 1 | 1 | 7 | 32 | 1 | 1 |
| 5 | 5 | 6 | 1 | 0 | 5 | 32 | 1 | 0 | 23 | 5 | 12 | 1 | 1 | 3 | 8 | 1 | 1 |
| 6 | 7 | 12 | 1 | 0 | 7 | 8 | 1 | 0 | 24 | 7 | 4 | 1 | 1 | 5 | 16 | 1 | 1 |
| 7 | 3 | 6 | 1 | 1 | 7 | 8 | 2 | 1 | 25 | 3 | 6 | 1 | 0 | 3 | 16 | 2 | 1 |
| 8 | 5 | 12 | 1 | 1 | 3 | 16 | 2 | 1 | 26 | 5 | 12 | 1 | 0 | 5 | 32 | 2 | 1 |
| 9 | 7 | 4 | 1 | 1 | 5 | 32 | 2 | 1 | 27 | 7 | 4 | 1 | 0 | 7 | 8 | 2 | 1 |
| 10 | 3 | 12 | 2 | 0 | 5 | 8 | 2 | 1 | 28 | 3 | 6 | 2 | 1 | 5 | 16 | 1 | 0 |
| 11 | 5 | 4 | 2 | 0 | 7 | 16 | 2 | 1 | 29 | 5 | 12 | 2 | 1 | 7 | 32 | 1 | 0 |
| 12 | 7 | 6 | 2 | 0 | 3 | 32 | 2 | 1 | 30 | 7 | 4 | 2 | 1 | 3 | 8 | 1 | 0 |
| 13 | 3 | 12 | 2 | 1 | 3 | 32 | 1 | 1 | 31 | 3 | 12 | 2 | 0 | 7 | 16 | 2 | 0 |
| 14 | 5 | 4 | 2 | 1 | 5 | 8 | 1 | 1 | 32 | 5 | 4 | 2 | 0 | 3 | 32 | 2 | 0 |
| 15 | 7 | 6 | 2 | 1 | 7 | 16 | 1 | 1 | 33 | 7 | 6 | 2 | 0 | 5 | 8 | 2 | 0 |
| 16 | 3 | 12 | 2 | 1 | 5 | 8 | 2 | 0 | 34 | 3 | 4 | 2 | 0 | 5 | 32 | 1 | 1 |
| 17 | 5 | 4 | 2 | 1 | 7 | 16 | 2 | 0 | 35 | 5 | 6 | 2 | 0 | 7 | 8 | 1 | 1 |
| 18 | 7 | 6 | 2 | 1 | 3 | 32 | 2 | 0 | 36 | 7 | 12 | 2 | 0 | 3 | 16 | 1 | 1 |

$$SS_T = \left[\sum_{i=1}^{N} y_i^2\right] - \frac{T^2}{N}, \tag{7}$$

$$SS_A = \left[\sum_{i=1}^{K_A}\left(\frac{A_i^2}{n_{A_i}}\right)\right] - \frac{T^2}{N}, \tag{8}$$

$$SS_e = SS_T - \left(SS_A + SS_B + \ldots\right), \tag{9}$$

$$MS_A = \frac{SS_A}{DF}, \tag{10}$$

$$MS_e = \frac{SS_{Error}}{DF}, \tag{11}$$

$$F_A = \frac{MS_A}{MS_e}, \tag{12}$$

$$PC(\%) = \frac{SS_A}{SS_T} \times 100, \tag{13}$$

where $K_A$ represents the number of levels of factor $A$, $SS_T$ represents $SS$ of the total change, $T$ represents the sum of all observations, $N$ represents the total number of experiments, $SS_A$ represents $SS$ for factor $A$, $A_i$ represents the sum of all observations at level $i$ for factor $A$, $n_{Ai}$ represents the number of observations at level $i$ for factor $A$, $MS_A$ represents the variance of factor $A$, $MS_e$ represents the variance in the error for factor $A$, and $F_A$ represents the $F$ ratio of factor $A$.

**Step 7**: Determine whether the goal has been achieved

In ANOVA, F indicates that a factor has a stronger influence. PC indicates that if the level of a factor changes, the degree of network parameter optimization may be affected. If the obtained control factors improve the accuracy of a CNN, the experiment is concluded; otherwise, the process returns to Step 3 to redetermine the levels of the control factors.

## 2.3 Materials

In this study, facial images from 66 people (63 women and 3 men) were collected to detect skin problems. For each face, 2–4 images of different parts of the skin were collected. The total number of images was 212, and the dataset was divided into three categories: normal condition, pigmented spots and wrinkles, and red skin and acne (Fig. 4). The images in this dataset were obtained using a charge-coupled device (CCD). The specifications of the optical sensor in this CCD are presented in Table 4. To increase the diversity of data, we increased the number of images in the original dataset. A total of 10 lighting levels between 0.6 and 1.5 eV (in increments of 0.1 eV) were used for each image to achieve a 10-fold increase in the number of images. The augmented dataset contained 2332 images. Table 5 presents the numbers of images before and after data augmentation.



Fig. 4.    (Color online) Example images from the dataset.

Table 4
Specifications of the optical sensor of the CCD.

| Vendor | Product model number | Reported resolution (MP) | Pixel pitch (μm) | Die length (mm) | Die width (mm) | Die area (mm$^2$) |
|---|---|---|---|---|---|---|
| Sony | A2111 | 12.0 | 1.00 | 4.04 | 5.53 | 22.3 |
| Sony | A2161 | 12.0 | 1.40 | 5.78 | 7.01 | 40.5 |

Table 5
Numbers of images before and after data augmentation.

| Classification | Number of images before increment | Number of images after increment |
|---|---|---|
| Normal | 46 | 506 |
| Pigmented spots plus wrinkles | 79 | 869 |
| Red skin plus acne | 87 | 957 |
| Total | 212 | 2332 |

## 3.    Experimental Results

### 3.1    Performance analysis

In this study, the L36 OA of the Taguchi method was used to optimize the parameters of the developed CNN. We then used the optimized network parameters to detect different skin conditions in the augmented dataset. We examined the adjustable factors that affect the accuracy of the CNN predictions and selected eight factors (the convolution kernel, filter, stride, and padding size in the two convolutional layers) as the control factors. To improve the reliability of the experimental results, we combined the test results obtained from a set of experiments using the orthogonal table to calculate the average observation results. Table 6 presents the experimental parameters, the accuracy of the CNN, and the S/N ratio for each set of experiments. S/N ratios were calculated using the ideal quality function, namely, the higher-the-better function. The larger the S/N ratio, the smaller the quality loss and the higher the quality obtained. According to the results obtained from 36 sets of experiments, the maximum mean accuracy (87.62%) was achieved in Run #5 and the minimum mean accuracy (74.52%) was achieved in Run #16.

The S/N ratio was improved by implementing better concepts. The mean S/N ratios obtained for each level and the optimal parameters obtained using the yield extraction method are presented in Table 7 and Fig. 5. The importance ranking of parameters was as follows: conv2_Filter > conv1_Stride > conv2_Stride > conv1_Kernel size > conv2_Padding > conv2_Kernel size > conv1_Filter > conv1_Padding. The optimal levels were $A2$ (level 2 for conv1_Kernel size), $B3$ (level 3 for conv1_Filter), $C1$ (level 1 for conv1_Stride), $D2$ (level 2 for conv1_Padding), $E3$ (level 3 for conv2_Kernel), $F3$ (level 3 for conv2_Filter), $G1$ (level 1 for conv2_Stride), and $H2$ (level 2 for conv2_Padding).

The ANOVA results presented in Table 8 indicate the degree of influence of each control factor and the optimized parameter combination. The PCs of factors F and C were 36.26 and 31.28%, respectively; thus, conv2_Filter and conv1_Stride had a high influence on the optimization results. The PCs of factors $B$, $D$, $E$, and $H$ were 0.76, 0.05, 0.87, and 2.4%, respectively; thus, conv2_Filter and conv1_Stride had a limited influence on the optimization results.

Table 6
S/N ratios for each set of experiments.

| Run | Factor | | | | | | | | Result | | | | |
|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | B | C | D | E | F | G | H | $Y_1$ (%) | $Y_2$ (%) | $Y_3$ (%) | $Y_{avg}$ (%) | S/N(Y) |
| 1 | 3 | 4 | 1 | 0 | 3 | 8 | 1 | 0 | 84.86 | 86.00 | 79.00 | 83.29 | −1.607 |
| 2 | 5 | 6 | 1 | 0 | 5 | 16 | 1 | 0 | 85.29 | 88.29 | 85.29 | 86.29 | −1.284 |
| 3 | 7 | 12 | 1 | 0 | 7 | 32 | 1 | 0 | 85.86 | 83.86 | 86.00 | 85.24 | −1.389 |
| 4 | 3 | 4 | 1 | 0 | 3 | 16 | 1 | 0 | 86.14 | 83.43 | 87.29 | 85.62 | −1.353 |
| 5 | 5 | 6 | 1 | 0 | 5 | 32 | 1 | 0 | 87.29 | 87.43 | 88.14 | 87.62 | −1.148 |
| 6 | 7 | 12 | 1 | 0 | 7 | 8 | 1 | 0 | 85.14 | 85.71 | 81.71 | 84.19 | −1.501 |
| 7 | 3 | 6 | 1 | 1 | 7 | 8 | 2 | 1 | 83.29 | 78.86 | 74.00 | 78.71 | −2.109 |
| 8 | 5 | 12 | 1 | 1 | 3 | 16 | 2 | 1 | 85.29 | 83.86 | 85.00 | 84.71 | −1.442 |
| 9 | 7 | 4 | 1 | 1 | 5 | 32 | 2 | 1 | 87.43 | 86.71 | 84.86 | 86.33 | −1.279 |
| 10 | 3 | 12 | 2 | 0 | 5 | 8 | 2 | 1 | 77.00 | 81.14 | 70.43 | 76.19 | −2.407 |
| 11 | 5 | 4 | 2 | 0 | 7 | 16 | 2 | 1 | 81.14 | 76.43 | 83.43 | 80.33 | −1.920 |
| 12 | 7 | 6 | 2 | 0 | 3 | 32 | 2 | 1 | 84.00 | 76.86 | 84.71 | 81.86 | −1.764 |
| 13 | 3 | 12 | 2 | 1 | 3 | 32 | 1 | 1 | 86.14 | 85.86 | 86.00 | 86.00 | −1.310 |
| 14 | 5 | 4 | 2 | 1 | 5 | 8 | 1 | 1 | 80.86 | 85.43 | 82.57 | 82.95 | −1.630 |
| 15 | 7 | 6 | 2 | 1 | 7 | 16 | 1 | 1 | 81.71 | 84.71 | 84.00 | 83.48 | −1.572 |
| 16 | 3 | 12 | 2 | 1 | 5 | 8 | 2 | 0 | 77.29 | 71.29 | 75.00 | 74.52 | −2.569 |
| 17 | 5 | 4 | 2 | 1 | 7 | 16 | 2 | 0 | 74.71 | 82.86 | 83.71 | 80.43 | −1.927 |
| 18 | 7 | 6 | 2 | 1 | 3 | 32 | 2 | 0 | 83.00 | 76.14 | 85.00 | 81.38 | −1.819 |
| 19 | 3 | 4 | 1 | 1 | 7 | 32 | 2 | 0 | 82.86 | 86.29 | 86.00 | 85.05 | −1.411 |
| 20 | 5 | 6 | 1 | 1 | 3 | 8 | 2 | 0 | 81.43 | 80.14 | 80.57 | 80.71 | −1.862 |
| 21 | 7 | 12 | 1 | 1 | 5 | 16 | 2 | 0 | 82.57 | 82.57 | 86.43 | 83.86 | −1.535 |
| 22 | 3 | 6 | 1 | 1 | 7 | 32 | 1 | 1 | 86.57 | 87.00 | 85.43 | 86.33 | −1.277 |
| 23 | 5 | 12 | 1 | 1 | 3 | 8 | 1 | 1 | 86.43 | 83.57 | 84.43 | 84.81 | −1.434 |
| 24 | 7 | 4 | 1 | 1 | 5 | 16 | 1 | 1 | 87.57 | 84.14 | 86.71 | 86.14 | −1.299 |
| 25 | 3 | 6 | 1 | 0 | 3 | 16 | 2 | 1 | 83.29 | 85.00 | 84.29 | 84.19 | −1.496 |
| 26 | 5 | 12 | 1 | 0 | 5 | 32 | 2 | 1 | 86.14 | 88.29 | 86.43 | 86.95 | −1.216 |
| 27 | 7 | 4 | 1 | 0 | 7 | 8 | 2 | 1 | 83.57 | 82.71 | 82.29 | 82.86 | −1.634 |
| 28 | 3 | 6 | 2 | 1 | 5 | 16 | 1 | 0 | 83.14 | 84.14 | 76.29 | 81.19 | −1.835 |
| 29 | 5 | 12 | 2 | 1 | 7 | 32 | 1 | 0 | 86.71 | 86.43 | 86.57 | 86.57 | −1.253 |
| 30 | 7 | 4 | 2 | 1 | 3 | 8 | 1 | 0 | 80.43 | 79.00 | 77.71 | 79.05 | −2.045 |
| 31 | 3 | 12 | 2 | 0 | 7 | 16 | 2 | 0 | 85.71 | 82.14 | 84.57 | 84.14 | −1.504 |
| 32 | 5 | 4 | 2 | 0 | 3 | 32 | 2 | 0 | 80.43 | 77.14 | 81.71 | 79.76 | −1.972 |
| 33 | 7 | 6 | 2 | 0 | 5 | 8 | 2 | 0 | 77.57 | 79.29 | 77.43 | 78.10 | −2.149 |
| 34 | 3 | 4 | 2 | 0 | 5 | 32 | 1 | 1 | 85.86 | 80.86 | 85.14 | 83.95 | −1.529 |
| 35 | 5 | 6 | 2 | 0 | 7 | 8 | 1 | 1 | 81.86 | 84.43 | 83.71 | 83.33 | −1.586 |
| 36 | 7 | 12 | 2 | 0 | 3 | 16 | 1 | 1 | 86.29 | 83.57 | 80.71 | 83.52 | −1.573 |

Table 7
Mean S/N ratios for each level and the optimal parameters.

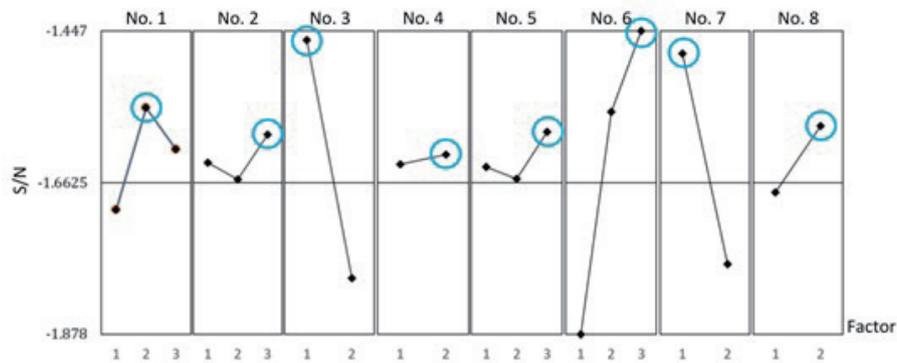| Factor | A | B | C | D | E | F | G | H |
|--------|---|---|---|---|---|---|---|---|
| Level 1 | −1.701 | −1.634 | −1.460 | −1.636 | −1.640 | −1.878 | −1.479 | −1.676 |
| Level 2 | −1.556 | −1.658 | −1.798 | −1.622 | −1.657 | −1.562 | −1.778 | −1.582 |
| Level 3 | −1.615 | −1.594 | — | — | −1.590 | −1.447 | — | — |
| Max−Min | 0.145 | 0.064 | 0.338 | 0.014 | 0.067 | 0.430 | 0.299 | 0.094 |
| Ranking | 4 | 7 | 2 | 8 | 6 | 1 | 3 | 5 |
| Best level | 2 | 3 | 1 | 2 | 3 | 3 | 1 | 2 |

Fig. 5.　(Color online) Response graph of the S/N ratios for different factor levels.

Table 8
ANOVA results.

| Factor | DOF | SS | $F_A$ | PC (%) |
|---|---|---|---|---|
| A | 2 | 0.1268 | 0.5496 | 3.86 |
| B | 2 | 0.0251 | 0.1030 | 0.76 |
| C | 1 | 1.0290 | 8.5778 | **31.28** |
| D | 1 | 0.0017 | 0.0161 | 0.05 |
| E | 2 | 0.0287 | 0.1175 | 0.87 |
| **F** | 2 | 1.1927 | 6.3989 | **36.26** |
| G | 1 | 0.8063 | 8.4403 | 24.51 |
| H | 1 | 0.0789 | 0.6746 | 2.40 |
| Sum | 35 | — | — | 100 |

## 3.2　Comparison of methods

　　We compared the accuracy of the Taguchi-optimized CNN model with that of the original CNN model. These CNN models were optimized using the optimal parameter configuration to increase their accuracies. The parameter settings and accuracies of these CNNs are presented in Table 9. After three additional experiments with the optimal parameter combination, the original CNN model and the CNN model with Taguchi parametric optimization yielded accuracies of 79.71 and 86.95%, respectively. Thus, the accuracy of the CNN with Taguchi parametric optimization was 7.24% higher than that of the original CNN. These results indicate that parametric optimization is essential for CNN models. The optimum kernel, filter, stride, and padding sizes were 5, 12, 1, and 1 for the first convolutional layer and 7, 32, 1, and 1 for the second convolutional layer, respectively.

　　We compared the facial image classification performance of the Taguchi-optimized CNN, VGG-16,[13] AlexNet,[12] and the original CNN. The classification results obtained with these models are presented in Table 10. Optimal recall, precision, FPR, and accuracy values of 89.1, 88.8, 11.2, and 86.95%, respectively, were obtained for the CNN with Taguchi parametric optimization. Moreover, the performance of the network improved after parameter optimization. In particular, the precision of class 1 increased to 100% for the CNN with Taguchi parametric optimization. The experimental results indicate that the proposed CNN is superior to the other networks.

Table 9
Comparison of the accuracies and parameter settings of the original CNN and the CNN with Taguchi parametric optimization.

| Layer | Factor | Original CNN | CNN with Taguchi parametric optimization |
|---|---|---|---|
| Conv 1 | Kernel size | 5 | 5 |
| | Filter | 6 | 12 |
| | Stride | 1 | 1 |
| | Padding | 0 | 1 |
| Conv 2 | Kernel size | 5 | 7 |
| | Filter | 16 | 32 |
| | Stride | 1 | 1 |
| | Padding | 0 | 1 |
| Accuracy (%) | | 79.71 | 86.95 |

Table 10
Comparison of the facial image classification performance.

| Method | | Recall (%) | | | Precision (%) | | | FPR (%) | | Accuracy (%) |
|---|---|---|---|---|---|---|---|---|---|---|
| VGG-16 | 1 | 82.9 | | 1 | 39.4 | | 1 | 60.6 | | 46.57 |
| | 2 | 50.5 | 53.0 | 2 | 54.7 | 50.8 | 2 | 45.3 | 49.2 | |
| | 3 | 25.7 | | 3 | 58.3 | | 3 | 41.7 | | |
| AlexNet | 1 | 72.4 | | 1 | 91.7 | | 1 | 8.3 | | 70.57 |
| | 2 | 69.3 | 70.9 | 2 | 67.5 | 74.7 | 2 | 32.5 | 25.3 | |
| | 3 | 70.9 | | 3 | 64.9 | | 3 | 35.1 | | |
| Original CNN | 1 | 86.2 | | 1 | 92.3 | | 1 | 7.7 | | 79.71 |
| | 2 | 74.6 | 79.8 | 2 | 77.3 | 80.9 | 2 | 22.7 | 19.1 | |
| | 3 | 78.5 | | 3 | 73.0 | | 3 | 27.0 | | |
| CNN with Taguchi parametric optimization | 1 | 100 | | 1 | 98.7 | | 1 | 1.3 | | **86.95** |
| | 2 | 82.2 | **89.1** | 2 | 86.1 | **88.8** | 2 | 13.9 | **11.2** | |
| | 3 | 85.1 | | 3 | 81.6 | | 3 | 18.4 | | |

## 4.    Conclusions

In this paper, a CNN with Taguchi parametric optimization was proposed for facial skin condition detection. We used the Taguchi method with the L36 OA and mixing stage factor VIII to increase the accuracy of facial skin condition detection. The ANOVA results indicated that the conv2_Filter and conv1_Stride factors had a high influence on the results. The original CNN model attained an accuracy of 79.71% and the average accuracy of the CNN with Taguchi parametric optimization was 86.95%; thus, Taguchi parametric optimization improved the accuracy by 7.24%. We also compared the classification performance of the Taguchi-optimized CNN, VGG-16, AlexNet, and the original CNN. Optimal recall, precision, FPR, and accuracy values of 89.1, 88.8, 11.2, and 86.95%, respectively, were obtained for the CNN with Taguchi parametric optimization. The experimental results indicate that the proposed hyperparameter optimization method can effectively improve the accuracy of facial skin condition detection.

We used the Taguchi method to find the optimal parameter combination for a CNN architecture to reduce the experiment time and increase the detection accuracy. A limitation of

this study is that only the lighting level was adjusted for data augmentation. In future studies, facial images can be divided into multiple areas, and the misclassification rate can be reduced by using other data augmentation methods, such as flipping, rotation, and cropping, and by developing new architectures for hyperparameter optimization. In addition, only three classes were chosen for skin detection in this study, and we will explore more classes in future work.

# References

1  Y. Zhai, Y. Huang, Y. Xu, J. Gan, H. Cao, W. Deng, R. D. Labati, V. Piuri, and A. F. Scotti: IEEE Access **8** (2020) 56892. https://doi.org/10.1109/ACCESS.2020.2980248

2  T. Igarashi, K. Nishino, and S. K. Nayar: Found. Trends Comput. Graph. Vis. **3** (2007) 1. https://doi.org/10.1561/0600000013

3  A. Ledoux and N. Richard: 2015 Conf. Cosmetic Measurements and Testing (IEEE, 2015). https://doi.org/10.1109/COMET.2015.7449662

4  S. Khan, N. Islam, Z. Jan, I. U. Din, and J. J. P. C. Rodrigues: Pattern Recognit. Lett. **125** (2019) 1. https://doi.org/10.1016/j.patrec.2019.03.022

5  J. Gan, L. Xiang, Y. Zhai, C. Mai, G. He, J. Zeng, Z. Bai, R. D. Labati, V. Piuri, and A. F. Scotti: IEEE Access **8** (2020) 20245. https://doi.org/10.1109/ACCESS.2020.2968837

6  C. J. Lin, C. H. Lin, and S. H. Jeng: Appl. Sci. **10** (2020) 3166. https://doi.org/10.3390/app10093166

7  H. Eun, D. Kim, C. Jung, and C. Kim: Comput. Put. Meth. Prog. Bio. **165** (2018) 215. https://doi.org/10.1016/j.cmpb.2018.08.012

8  B. Chen, C. Chen, and W. Hsu: IEEE Trans. Multimed **6** (2015) 804. https://doi.org/10.1109/TMM.2015.2420374

9  C. J. Lin, S. Y. Jeng, and M. K. Chen: Appl. Sci. **10** (2020) 2591. https://doi.org/10.3390/app10072591

10  L. Ningbo, X. Yanan, T. Yongha, M. Hongwei, and W. Shuliang: Future Gener Comput Syst. **94** (2019) 524. https://doi.org/10.1016/j.future.2018.11.036

11  Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner: Proc IEEE Inst. Electr. Electron. Eng. **86** (1998) 2278–2324. https://doi.org/10.1109/5.726791

12  A. Krizhevsky, I. Sutskever, and G. E. Hinton: Commun. ACM **60** (2017) 84. https://doi.org/10.1145/3065386

13  J. Wei, Y. Ibrahim, S. Qian, H. Wang, G. Liu, Q. Yu, R. Qian, and J. Shi: Microelectron. Reliab. **110** (2020) 113648. https://doi.org/10.1016/j.microrel.2020.113648

14  C. C. J. Kuo: J. Vis. Commun. Image Represent. **41** (2016) 406. https://doi.org/10.1016/j.jvcir.2016.11.003

15  C. J. Lin, S. Y. Jeng, and C. L. Lee: Diagnostics **10** (2020) 662. https://doi.org/10.3390/diagnostics10090662

16  S. Suresh and S. Mohan: J. King Saud. Univ. Comput. Inf. Sci. (2019) (in press) https://doi.org/10.1016/j.jksuci.2019.11.013

17  N. Biçer, T. Engin, H. Yaşar, E. Büyükkaya, A. Aydın, and A. Topuz: Int. J. Therm. Sci. **155** (2020) 106417. https://doi.org/10.1016/j.ijthermalsci.2020.106417

18  A. N. Özakın and F. Kaya: Sol. Energy **197** (2020) 199. https://doi.org/10.1016/j.solener.2019.12.077

19  R. C. Mahajan, V. Vyas, and M. S. Sutaone: Microelectron. Eng. **219** (2020) 111156. https://doi.org/10.1016/j.mee.2019.111156

20  V. Ayhan, Ç. Çangal, İ. Cesur, A. Çoban, G. Ergen, Y. Çay, A. Kolip, and İ. Özsert: Fuel **261** (2020) 116371. https://doi.org/10.1016/j.fuel.2019.116371

21  S. Özel, E. Vural, and M. Binici: Fuel **263** (2020) 116537. https://doi.org/10.1016/j.fuel.2019.116537

22  I. Düntsch and G. Gediga: Int. J. Approx. Reason **118** (2020) 115. https://doi.org/10.1016/j.ijar.2019.12.008

23  A. Özakın and F. Kaya: Sol. Energy **197** (2020) 199. https://doi.org/10.1016/j.solener.2019.12.077

24  F. N. Idris, M. M. Nadzir, and S. R. A. Shukor: J. Environ. Chem. Eng. **8** (2020) 103766. https://doi.org/10.1016/j.jece.2020.103766

## About the Authors

**Hsueh-Yi Lin** received his M.S. degree in computer science and information engineering from Northrop University, CA, USA, in 1989. Currently, he is an associate professor of the Computer Science and Information Engineering Department, National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C. His current research interests are machine learning, intelligent control, pattern recognition, and image processing.

**Cheng-Jian Lin** received his B.S. degree in electrical engineering from Ta Tung Institute of Technology, Taipei, Taiwan, R.O.C., in 1986, and his M.S. and Ph.D. degrees in electrical and control engineering from National Chiao-Tung University, Taiwan, R.O.C., in 1991 and 1996, respectively. Currently, he is a chair professor of the Computer Science and Information Engineering Department, National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C., and Dean of Intelligence College, National Taichung University of Science and Technology, Taichung, Taiwan, R.O.C. His current research interests are machine learning, pattern recognition, intelligent control, image processing, intelligent manufacturing, and evolutionary robots. (cjlin@ncut.edu.tw)

**Shiou-Yun Jeng** received her Ph.D. degree in industrial engineering and management from National Yunlin University of Science & Technology, Taiwan, R.O.C., in 2019. Currently, she is a postdoctoral research fellow of the Computer Science and Information Engineering Department, National Chin-Yi University of Technology, Taichung City, Taiwan, R.O.C. Her current research interests are artificial intelligence, fuzzy theory, sustainable supply chain management, resource efficiency, green production management, and big data analysis.

**Cheng-Yi Yu** received his Ph.D. degree in electrical engineering from National Chung Hsing University, Taichung, Taiwan, R.O.C., in 2011. Currently, he is an associate professor of the Computer Science and Information Engineering Department, National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C. His current research interests are machine learning, pattern recognition, and image processing.