


Article

Face Liveness Detection Using Thermal Face-CNN with External Knowledge

Jongwoo Seo ¹  and In-Jeong Chung ^{2,*}

¹ Department of Computer and Information Science, Korea University, Sejong Campus, Sejong City 30019, Korea; sjw007s@korea.ac.kr

² Department of Computer Convergence Software, Korea University, Sejong Campus, Sejong City 30019, Korea

* Correspondence: chung@korea.ac.kr

Received: 1 January 2019; Accepted: 6 March 2019; Published: 10 March 2019



Abstract: Face liveness detection is important for ensuring security. However, because faces are shown in photographs or on a display, it is difficult to detect the real face using the features of the face shape. In this paper, we propose a thermal face-convolutional neural network (Thermal Face-CNN) that knows the external knowledge regarding the fact that the real face temperature of the real person is 36~37 degrees on average. First, we compared the red, green, and blue (RGB) image with the thermal image to identify the data suitable for face liveness detection using a multi-layer neural network (MLP), convolutional neural network (CNN), and C-support vector machine (C-SVM). Next, we compared the performance of the algorithms and the newly proposed Thermal Face-CNN in a thermal image dataset. The experiment results show that the thermal image is more suitable than the RGB image for face liveness detection. Further, we also found that Thermal Face-CNN performs better than CNN, MLP, and C-SVM when the precision is slightly more crucial than recall through F-measure.

Keywords: face liveness detection; convolutional neural network; thermal image; external knowledge

1. Introduction

Face liveness detection in indoor residential environments is an important technique for delivering security information, such as in the case of unlocking a mobile device using a face recognition system. For example, in order to allow access to only one specific person, that person's unique information, such as their face, can be used to unlock security measures. However, because the printed face photograph and face from the display can sufficiently generate the unique information of the face, the reliability of the security is reduced. Therefore, there is a need to provide more secure security by using face liveness detection, in which thermal images are distinguishable between the real face and the fake face through the heat distribution existing in the face of the real person.

In this paper, we first quantitatively identify a more suitable image for face liveness detection using both the RGB image and the thermal image. The same algorithms were applied to the RGB and thermal image datasets for the comparison. A multi-layer neural network (MLP) [1], convolutional neural network (CNN) [2], and C-support vector machine (C-SVM) [3] with a smooth hyperplane were used for the comparison. In addition, we compared the performance of the existing algorithms with thermal face-convolutional neural network (Thermal Face-CNN) proposed in this paper. Thermal Face-CNN is an algorithm with external knowledge about the temperature values that are found in a real face.

We have collected thermal images because there are many RGB image datasets for face liveness detection but few or no thermal image datasets available. We obtained RGB and thermal images

of the same scene in order to evaluate how these thermal images improve performance over RGB images. Accuracy [4], recall [4], and precision [4] were mainly obtained on both the RGB and thermal image datasets.

The experimental results show that the best-performing CNN performance has an accuracy of 0.6898, a recall of 0.5752, and a precision of 0.7342 on the RGB image dataset, while it has an accuracy of 0.8367, a recall of 0.7876, and a precision of 0.8476 on the thermal image dataset. Therefore, it has been shown that the thermal image is more effective in face liveness detection than the RGB image. In addition, we show that the average recall value is improved by 13.72% over CNN by using the Thermal Face-CNN proposed in this paper for the thermal image dataset. It is also shown that we found that Thermal Face-CNN performs better than CNN, MLP, and C-SVM when the precision is slightly more crucial than recall through F-measure.

2. Background and Related Work

Face detection is a field involving the detection of a face in an image. Algorithms for face detection judge whether or not the object in the picture is the face [5]. However, face liveness detection is a field in which the face presented is judged to be the real face or the fake face or no face. Therefore, face detection is a very different field from face liveness detection. For this reason, a paper related to face detection could not be compared with a paper related to face liveness detection. In the field of face liveness detection, there are three ways to imitate a real face: using a picture with that face, replaying a video with that face, and using a 3D face mask [6]. The method using the picture with the face involves printing the face on paper or displaying the face on a display. In order to solve this problem, studies have been carried out to explore ways to detect the real face using a photo-based dataset [6–9]. In addition, there have been studies into the use of video-based datasets to distinguish the real face from the fake face [7,10]. Further studies into ways to distinguish between the real face and the 3D face mask have also been conducted [11,12].

Many datasets can be used for face liveness detection: NUAA [8], ZJU Eyeblink [13], Idiap Print-attack [14], Idiap Replay-attack [10], CASIA FASD [15], MSU-MFSD [16], MSU RAFS [17], UVAD [18,19], MSU USSA [6], and so on. However, these datasets include data composed of RGB images. There are not enough datasets composed of thermal images. Therefore, research on face liveness detection with thermal images has been insufficient to date. Thermal images have already been used in research for face detection and pedestrian detection [20–23]. Thermal images can be obtained through the distribution of infrared rays, even at night when there is no visible light. Because RGB images have the disadvantage of being affected by the intensity of visible light, while thermal images have the advantage of being usable in places where there is no visible light, thermal images have been successfully applied in various fields. Therefore, it is necessary to compare the RGB image and the thermal image with regard to how much performance improvement is offered by the use of the thermal image in face liveness detection. For comparison, using an existing dataset would be ideal, but none of these contain information about temperature. Thus, a new dataset is needed.

Face liveness detection involves detecting the real face by analyzing the information obtained from the image. Therefore, previous studies on face liveness detection have been carried out using image processing methods. The support vector machine (SVM) is a classification algorithm that has been used to distinguish between the real and fake faces in face liveness detection [7,11]. As shown in these studies, SVM performs well in the area of classification. Of the SVM algorithms, the linear SVM finds the linear hyperplane with the largest margin [24]. The linear SVM assumes that classification can be performed by a line. However, there are cases where the data to be classified cannot be simply classified as a line. In order to solve this problem, research was carried out on nonlinear SVM using kernel functions [24]. The classification was proceeded using SVM on the abstraction information combining static features and dynamic features for face liveness detection in [7]. In addition, SVM learned the multispectral reflectance distribution information that can distinguish real human skin from images or objects meant to look like skin for face liveness detection in [11]. Previously, SVM

used in face liveness detection learned to perfectly classify training data without error. However, there is another way to find a soft margin hyperplane that has the largest margins while allowing exceptional misclassification of the small amount of data in the learning data [3]. By using a soft margin hyperplane, we can find a hyperplane that is more generalizable without having an overfitting hyperplane on the learning data. Therefore, C-SVM, which is a nonlinear SVM using a soft margin hyperplane and more generalizable than the SVMs used in previous studies, was used in Section 4 to evaluate the performance of algorithms on the thermal image dataset.

The artificial neural network imitates human neurons [1]. In particular, MLP is one of the artificial neural networks used in image processing [25]. Image processing can be done through MLP, in which the information of pixels is inserted into the input layer, and the output layer outputs 0 and 1 with one node for binary classification. CNN [2], which is designed for effective image processing, is an algorithm that modifies MLP in a way that reduces weights and shares weights. There are studies that have effectively performed face liveness detection using CNN on the RGB image [7,26,27]. In addition, it is known that CNN is a more powerful algorithm for face liveness detection on the RGB image than SVM [26]. Furthermore, CNN can achieve 98.99% accuracy on the relatively easy RGB image dataset called NUAA [8], which means that CNN is superior to previous methods [26] and is state-of-the-art. An accuracy of 98.99% does not mean that this field is entirely conquered. There is a need to study more difficult face liveness detection by allowing multiple objects to be included simultaneously in an image and increasing a lot of computation with more pixels in an image. The thermal image can be used to do this because there have also been studies showing that CNN has been successfully used on the thermal image [20–22]. For these reasons, and because there is a need to properly process the thermal image used for face liveness detection with CNN, we used this algorithm in Section 4. Nevertheless, it is necessary to investigate an algorithm superior to CNN for face liveness detection based on the thermal image. The CNN algorithm and Thermal Face-CNN for face liveness detection are concretely described in Section 3 of this paper.

In addition to the support vector machine and the artificial neural network, the algorithms used for face liveness detection are diverse. A logistic regression model [8,28] was used to classify the real face and the fake face. In addition, as methods to identify the features of the image, local binary pattern [9,29] and Lambertian model [8] were used for face liveness detection. The local binary pattern is a method of extracting the feature of the image considering the difference of value relative to neighboring pixels on the basis of a pixel. By this method, the feature vector representing the feature of the image was extracted for face liveness detection [9]. Similarly, the Lambertian model is a method that has been studied for extracting information about the difference between the real face and fake face. Therefore, we can know that there has been a lot of research on how to extract image feature information in the related studies.

3. The Proposed Method

The proposed Thermal Face-CNN is an algorithm for face liveness detection based on CNN. In this algorithm, external knowledge for face liveness detection is inserted first, followed by CNN. In the proposed method, the artificial neural network part is the same as the existing CNN. CNN combines the convolutional layer, the pooling layer, and the fully connected layer. The number of convolutional layers, pooling layers, and fully connected layers vary depending on the number and type of pixels in the image. For visual convenience, an example of Thermal Face-CNN with two convolutional layers, two pooling layers, and one hidden layer is shown in Figure 1. The numbers of layers used are explained in Section 4.

The process of inserting external knowledge for face liveness detection

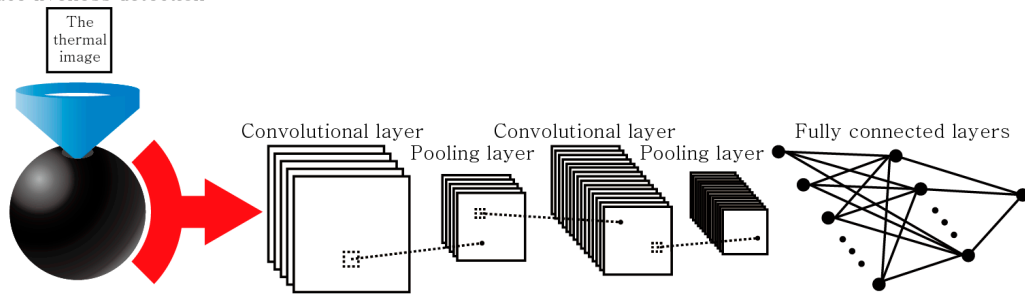


Figure 1. Thermal face-convolutional neural network (Thermal Face-CNN).

First, knowledge is inserted for face liveness detection. After that, the data with external knowledge is calculated in the convolutional layer and transferred to the pooling layer. This can be repeated several times in order to process the complex image. Next, CNN passes the previously obtained information to the fully connected layer. Finally, CNN classifies the image in the output layer. The process of inserting external knowledge, the convolutional layer, the pooling layer, and fully connected layer are explained as the paper continues. The process of inserting external knowledge for face liveness detection can be accomplished by the process of inserting knowledge about the temperature that a human face can have. This can be represented as Equation (1).

$$h = \begin{cases} \text{knowledge value} \times g & \text{if } \text{down limit} \leq g \leq \text{up limit} \\ g & \text{Otherwise} \end{cases} \quad (1)$$

In Equation (1), g is the measured temperature value, and h is the input value to CNN. Equation (1) is a formula that multiplies the value between *up limit* and *down limit* by *knowledge value* so as to make use of the physiological knowledge of the mean body temperature of a person, which is between 36 and 37 degrees [30]. A pixel measuring a part of a real face must have a temperature value in this vicinity. The fact that there is a high probability that a pixel with a value close to 36 or 37 degrees in a measured thermal image is likely to represent a part of a real face can only be obtained from external knowledge, not from the data. In order to insert this knowledge into the artificial neural network, we make a remarkably different value than the measured value using Equation (1). In this case, the artificial neural network recognizes the temperature of this pixel as very different from the temperature measured at other pixels. If the *knowledge value* is 10, it is about ten times larger than the values of other pixels. Figure 2 shows an example of selecting 34 and 39 values near the human body temperature of 36 and 37 degrees, taking into account the errors that may occur during measurement. In Section 4, we conducted experiments setting various values of *knowledge value*, *up limit*, and *down limit*.

In the graph shown in the upper left of Figure 2, the vertical axis represents the temperature values. In the graph shown in the upper right of Figure 2, the external knowledge about the possibility that a part of an object measured by each pixel is a part of a real face and the possibility that it is not is expressed. Note that there are no quantitative values in the vertical axis shown in the upper right graph in Figure 2. All of the graphs of the horizontal axes shown in Figure 2 represent the pixel index. In the upper left graph in Figure 2, pixels 2 and 3 are data with different meanings from the graph on the upper right, but there is almost no quantitative difference. In order to emphasize this content, input data must be re-expressed so that there are distinct differences between the two different data: one might measure a part of a real face, and the other might not. To do so, *knowledge value* in Equation (1) is used. As shown in the graph in Figure 2, below, information is forced to be distributed in a specific region through a considerable difference between real values, and thermal information about the temperature value of the pixels measured is also expressed showing a minute difference. The differences in measured temperatures can be seen by comparing pixel 1 to pixel 3 and pixel 2 to pixel 4. The optimal knowledge value can be empirically found through experimentation.

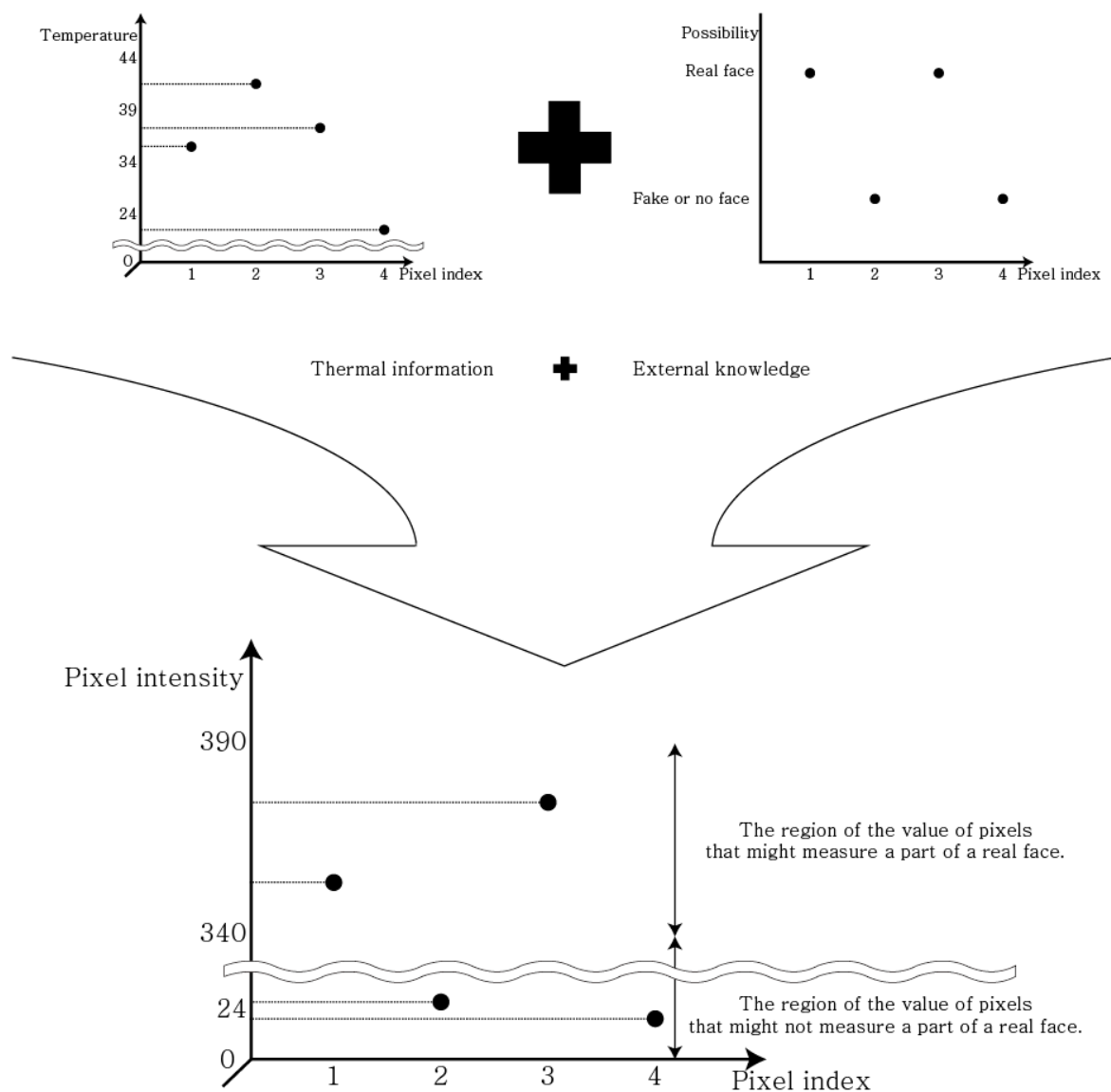


Figure 2. Example of the process of inserting external knowledge.

The convolutional layer serves to extract the complex features of the two-dimensional image [31]. The parameters of the convolutional layer are *kernel_size*, *filters*, and *stride*. *kernel_size* indicates the width and height of a kernel composed of learnable weights. *filters* represent the number of kernels, and *stride* is a parameter for extracting the characteristics of an image based on a certain interval. From the convolutional layer, we can extract the spatial information while sharing the weights [2]. Formal equations related to the convolutional layer are presented in [31]. The information calculated in the convolutional layer is transferred to the pooling layer.

Among the layers that make up CNN, the pooling layer induces spatial invariance by reducing the size of the feature map [32]. The parameters of the pooling layer are *pooling_size* and *stride*. *pooling_size* represents the size of the zone to be examined, such as *kernel_size*, a parameter of the convolutional layer discussed above. *stride* in the pooling layer serves the same purpose as the *stride* parameter of the convolutional layer. The max pooling layer has a function to find the maximum value in each region and to transfer it to the next layer [32]. Finally, the information is transferred to the fully connected layer through the convolutional layer and the pooling layer.

The fully connected layer is a type of layer used in MLP consisting of nodes completely connected to the nodes in each of the previous and subsequent layers [1].

4. Experiments

4.1. Data Collection and Experimental Environment Construction

The Flir C3 was used as the camera for collecting data. The camera has two lenses on the front: an RGB lens to obtain RGB images of 640×480 pixels and an infrared lens to obtain thermal images of 80×60 pixels. The information on the Flir C3 can be found at a website listed in Supplementary Materials at the end of this paper. We collected one RGB image and one thermal image in each scene to find suitable data for face liveness detection. Since a thermal image is better than an RGB image at night, we took images in indoor residential environments with visible light for accurate performance comparison. There were no conditions for the distance of the object. The faces in the dataset were used with and without a variety of accessories, such as glasses. No matter what, the face is covered by any object, which can cover anything except the eyes, nose, and mouth. We used the function of the Flir C3 that allows for the simultaneous operation of the two lenses. A total of 844 scenes were taken. The actual data used were 844 Excel files with temperature information collected from infrared lens and 2532 Excel files with R, G, and B information collected from RGB lens. In Figure 3, the images in the top row are RGB images, while the images in the bottom row are thermal images.

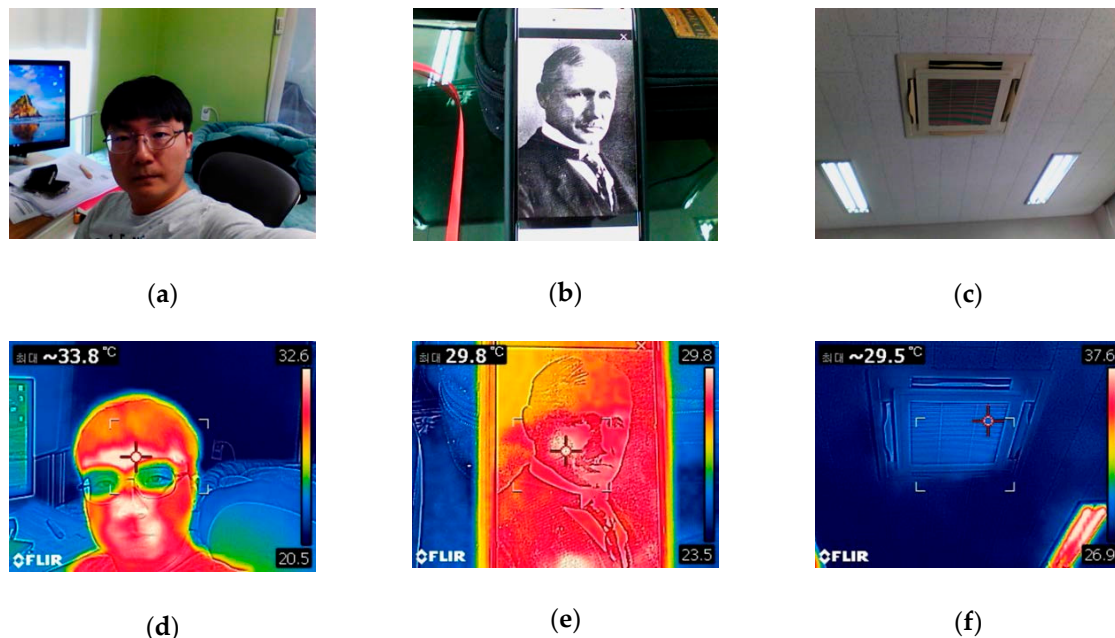


Figure 3. Data examples: (a) a real face taken by RGB lens; (b) a face on a display taken by RGB lens; (c) a ceiling air conditioner taken by RGB lens; (d) a real face taken by infrared lens; (e) a face on a display taken by infrared lens; (f) a ceiling air conditioner taken by infrared lens.

Figure 3a,d are RGB and thermal images with a real face present, respectively. Figure 3b,e are RGB and thermal images with a face on a display, respectively. Figure 3c,f shows images taken of a ceiling air conditioner with no face. In the thermal images, the color is obtained by the software in the thermal camera itself so that the measured temperature can be intuitively grasped visually. In Figure 3a,b,d,e, it can be seen that the outline of the heat distribution and the heat on the face from the display differ from those of the real face. The RGB face liveness detection dataset jongwoo (RFLDDJ) we created and the thermal face liveness detection dataset jongwoo (TFLDDJ) we created are available on the internet. In NUAA [8], the whole picture is completely filled with faces. However, in the RGB dataset we created, people and objects were shot in indoor living environments in order to increase the level of difficulty. In other words, multiple objects coexist in a single image in the datasets we made. The data

are more difficult because a more general situation is assumed. The information of the datasets can be found at websites listed in the Supplementary Materials at the end of this paper.

The numbers of pixels differ between the two lenses. The RGB lens has 640 pixels horizontally and 480 pixels vertically, for a total of 307,200 pixels on an image. By contrast, the infrared lens has 80 pixels horizontally and 60 pixels vertically, for a total of 4800 pixels on an image. The numbers of pixels in images obtained by the two lenses differ by 64 times. However, the range of actually measured scenes is not much different. Figure 4 shows its example.



Figure 4. Comparison of the ranges of lenses.

As shown in Figure 4, the number of pixels has a difference of 64 times, but there is not much difference in the area to be taken. In addition, because the RGB lens and the infrared lens have different pixel sizes, and because there is a slight difference in the position of each lens on the camera, it is not clear how many pixels from the horizontal, vertical, top, and bottom sides should be cut for the same range of the scene. Therefore, it is impossible to capture the same extent of the range of the scene. For the correct experiment, if the real face is in a scene that the infrared lens cannot capture as an image, this image was removed from the experiment.

We use Adam [33], Dropout [34], and ReLu [35] to improve learning abilities when learning CNN and Thermal Face-CNN. The Adam algorithm reduces error by learning the weights existing in the artificial neural network. It is easier to execute than the back-propagation algorithm [36]. It is also more efficient and requires less memory [33]. Dropout prevents overfitting by allowing each node not to participate in the calculation randomly during the learning process [34]. Sigmoid [37] was used as an activation function in the output layer of all artificial neural networks used in the experiments except for C-SVM, and ReLu was used as an activation function of the hidden layer. As the pooling layer, the max pooling layer [32] is used. In addition, the probability of dropping each node is 10%. An intel core i7-7820X CPU was used as the hardware in the experiment, and the memory was DDR4 32G. The experiment was carried out using the Tensorflow [38] library, which has artificial neural network code. In the case of C-SVM, the sklearn.svm.svc library was used to carry out the experiment. The information of the library can be found at a website listed in the Supplementary Materials at the end of this paper.

Accuracy [4], recall [4], and precision [4] were mainly used as evaluation indices in the experiment. In this study, accuracy refers to how the actual value and predicted value are matched, regardless of the presence or absence of a real face. Recall is an index of how many images having the real face are judged to have the real face. Precision is also an index of how many images have the real face among those predicted to have the real face.

4.2. The Comparison of Face Liveness Detection between the RGB Image and Thermal Image

Before examining the performance of the proposed Thermal Face-CNN, we obtained accuracy, recall, and precision for each RGB image and thermal image dataset in order to identify the appropriate dataset for face liveness detection. For the comparison, we used CNN, MLP, and C-SVM. The left side of Table 1 shows the parameters of CNN applied to the RGB image dataset, and the right side of

Table 1 shows the parameters of CNN applied to the thermal image dataset. We empirically sought the values of the parameters that would make the error of the artificial neural network converge to zero.

Table 1. Convolutional neural network (CNN) parameters used in the RGB image dataset and the thermal image dataset.

Parameter	Kernel_Size	Filters	Pool_Size	Stride/ Nodes	Parameter	Kernel_Size	Filters	Pool_Size	Stride/ Nodes
1st con_	(15, 15)	150	N/A	(3, 3)	1st con_	(20, 20)	50	N/A	(3, 3)
1st pool_	N/A	N/A	(5, 5)	(1, 1)	1st pool_	N/A	N/A	(3, 3)	(2, 2)
2nd con_	(15, 15)	130	N/A	(3, 3)	2nd con_	(5, 5)	30	N/A	(1, 1)
2nd pool_	N/A	N/A	(5, 5)	(1, 1)	2nd pool_	N/A	N/A	(2, 2)	(1, 1)
3rd con_	(15, 15)	100	N/A	(2, 2)	input_	N/A	N/A	N/A	1920
3rd pool_	N/A	N/A	(3, 3)	(1, 1)	hidden_	N/A	N/A	N/A	120
4th con_	(5, 5)	80	N/A	(2, 2)	output_	N/A	N/A	N/A	1
4th pool_	N/A	N/A	(2, 2)	(1, 1)	N/A	N/A	N/A	N/A	N/A
input_	N/A	N/A	N/A	1920	N/A	N/A	N/A	N/A	N/A
1st hidden_	N/A	N/A	N/A	1536	N/A	N/A	N/A	N/A	N/A
2nd hidden_	N/A	N/A	N/A	1200	N/A	N/A	N/A	N/A	N/A
3rd hidden_	N/A	N/A	N/A	1000	N/A	N/A	N/A	N/A	N/A
output_	N/A	N/A	N/A	1	N/A	N/A	N/A	N/A	N/A

In Table 1, *nodes* refers to the number of nodes in the corresponding layer. Further, *con_* means convolutional layer and *pool_* means pooling layer. *input_*, *hidden_*, and *output_* mean input layer, hidden layer, and output layer, respectively. The rest of the parameters are the same as those described in Section 3. In Table 1, the values in parentheses represent two values for the width and length of the kernel and pooling sequentially.

The parameter values for C-SVM used in the thermal image dataset are shown in Table 2.

Table 2. C-support vector machine (C-SVM) parameters used in the thermal image dataset.

Parameter	Error Penalty	Kernel	Gamma	Tolerance	Degree
Value	c	RBF or POLY	$1/n_Features$	0.001	3

In Table 2, c is an *error penalty* parameter, and we changed c when we experimented. RBF [39] or polynomial (POLY) [39] is used as *kernel*. γ is the coefficient of *kernel*. In addition, $n_features$ means the number of features and *tolerance* means stopping criterion. *degree* means the degree of the polynomial kernel function.

The parameters of the MLP used to learn the thermal images are shown in Table 3.

Table 3. Multi-layer neural network (MLP) parameters in the thermal image dataset.

Parameter	Input_	1st Hidden_	2nd Hidden_	3rd Hidden_	4th Hidden_	Output_
Nodes	4800	3000	2000	1500	1000	1

A total of 599 images in the RGB image dataset and thermal image dataset from image 1 to image 599 were used as training data, and the remaining 245 images were used for test data. There are 338 images of 844 images with the real face, and 506 images without the real face. In the training set are 225 images with the real face, and 113 images with the real face are in test set. In the training set were 374 images without the real face, and 132 images without the real face are in the test set. Table 4 shows the experimental results of CNN in the RGB image dataset and the thermal image dataset. Tables 5 and 6 show the experimental results of MLP and C-SVM in the thermal image dataset. The figures in the following tables, including Tables 4–6, were rounded to the fourth decimal place. Figures expressed as percentages in the following tables were rounded to the second decimal place.

Table 4. CNN's performance in the RGB image dataset and the thermal image dataset.

Index	In the RGB Image Dataset			Index	In the Thermal Image Dataset		
	Accuracy	Recall	Precision		Accuracy	Recall	Precision
Average	0.658	0.4779	0.6871	Average	0.7816	0.6996	0.8022
The best	0.6898	0.5752	0.7342	The best	0.8367	0.7876	0.8476

Table 5. MLP's performance in the thermal image dataset.

Index	MLP		
	Accuracy	Recall	Precision
Average	0.7551	0.4991	0.9431
The best	0.7837	0.5664	0.9524

Table 6. C-SVM's performance in the thermal image dataset.

kernel	c	Accuracy	Recall	Precision	Kernel	c	Accuracy	Recall	Precision
RBF	0.7	0.5429	0.0088	1	POLY	0.06	0.7388	0.6195	0.7692
	0.8	0.8163	0.9381	0.7361		0.07	0.7388	0.6195	0.7692
	0.81	0.8082	0.9381	0.726		0.07	0.7388	0.6195	0.7692
	0.82	0.8204	0.9646	0.7315		0.08	0.7388	0.6195	0.7692
	0.83	0.8204	0.9646	0.7315		0.08	0.7388	0.6195	0.7692
	0.84	0.8122	0.9646	0.7219		0.09	0.7388	0.6195	0.7692
	0.85	0.8082	0.9646	0.7171		0.1	0.7388	0.6195	0.7692
	0.86	0.8082	0.9646	0.7171		0.11	0.7388	0.6195	0.7692
	0.87	0.8082	0.9646	0.7171		0.13	0.7388	0.6195	0.7692
	0.88	0.8082	0.9646	0.7171		0.14	0.7388	0.6195	0.7692
	0.89	0.8122	0.9646	0.7219		0.17	0.7388	0.6195	0.7692
	0.9	0.8122	0.9646	0.7219		0.2	0.7388	0.6195	0.7692
	0.91	0.8163	0.9646	0.7267		0.25	0.7388	0.6195	0.7692
	0.92	0.8204	0.9646	0.7315		0.3	0.7388	0.6195	0.7692
	0.93	0.8204	0.9646	0.7315		0.33	0.7388	0.6195	0.7692
	0.94	0.8204	0.9646	0.7315		0.4	0.7388	0.6195	0.7692
	0.95	0.8122	0.9469	0.7279		0.5	0.7388	0.6195	0.7692
	0.96	0.8122	0.9381	0.731		0.6	0.7388	0.6195	0.7692
	0.97	0.8163	0.9381	0.7361		0.7	0.7388	0.6195	0.7692
	0.98	0.8163	0.9381	0.7361		0.8	0.7388	0.6195	0.7692
	0.99	0.8204	0.9381	0.7413		0.9	0.7388	0.6195	0.7692
POLY	1	0.8245	0.9381	0.7465		1	0.7388	0.6195	0.7692
	1.5	0.8204	0.9292	0.7447		1.5	0.7388	0.6195	0.7692
	2	0.8204	0.9292	0.7447		2	0.7388	0.6195	0.7692
	2.5	0.8204	0.9292	0.7447		2.5	0.7388	0.6195	0.7692

In Tables 4 and 5, “The best” refers to the highest values. “Average” means the average value. In order to obtain the information shown in Table 4, five CNNs in the RGB image dataset and 20 CNNs in the thermal image dataset were implemented with the same parameters. Because the combinations of weights obtained when the neural network is learned with the same parameters are always different and show different performances, we repeated the experiment 20 times in order to obtain the average performance of the general accuracy, recall, and precision values. However, in the RGB image dataset, the number of pixels contained in each image was 907,200, which required a substantial amount of computation. Therefore, 20 CNNs were learned in the thermal image dataset, but only five CNNs were learned in the RGB image dataset. To obtain Table 5, five MLPs were learned because MLP requires a large amount of computation. To evaluate C-SVM's performance in Table 6, we obtained one C-SVM on each parameter setting. The values of accuracy, recall, and precision shown in Table 4, which were obtained using the thermal image dataset, are higher than those of the RGB image dataset. It can be seen from the above that, on CNN, the thermal image is more suitable than the RGB image.

In the case of MLP, since there is 907,200-pixel information per RGB image, the number of nodes in the input layer should also be 907,200. We tried to implement an MLP with about 900,000 nodes

in the input layer, but the hardware limitations made it impossible to calculate. Further, the C-SVM was learned using the parameters shown in Table 2, but it was determined that there was no real face for all the test data, because it was not learned properly. However, as shown in Tables 5 and 6, MLP and C-SVM can be learned because of the small number of pixels in a thermal image data. Through comparing Tables 4–6, it can be seen that good performance can be obtained by the thermal image data.

4.3. Performance Comparison of CNN, C-SVM, and Thermal Face-CNN

Section 4.2 showed that the thermal image is better than the RGB image. In Section 4.3, we applied the Thermal Face-CNN proposed in this paper to the thermal image with superior performance for face liveness detection than the RGB image, and we compared its performance with those of the other algorithms. We used the same parameters of CNN on the thermal image dataset for Thermal Face-CNN. We also constructed 20 Thermal Face-CNNs with the same parameter setting as used in the experiment on 20 CNNs, shown in Table 4. The accuracy, recall, and precision values of Thermal Face-CNNs are shown in Tables 7–12. Parenthetical values in these tables indicate *knowledge value*, *up limit*, and *down limit* values, sequentially.

Table 7. Thermal Face-CNN accuracy, recall, and precision values 1.

Index	(10, 39, 34)			Index	(10, 40, 34)		
	Accuracy	Recall	Precision		Accuracy	Recall	Precision
Average	0.7967	0.7726	0.784	Average	0.7957	0.7602	0.7901
The best	0.8245	0.8584	0.8173	The best	0.8204	0.8407	0.8235
A_im (%)	1.93	10.44	−2.27	A_im (%)	1.77	7.97	−1.53
M_im (%)	−1.46	8.99	−3.58	M_im (%)	−1.99	6.32	−2.93

Table 8. Thermal Face-CNN accuracy, recall, and precision values 2.

Index	(10, 41, 34)			Index	(10, 39, 35)		
	Accuracy	Recall	Precision		Accuracy	Recall	Precision
Average	0.7894	0.7491	0.7851	Average	0.7929	0.7385	0.7986
The best	0.8286	0.8142	0.8286	The best	0.8327	0.8142	0.8391
A_im (%)	0.99	6.61	−2.18	A_im (%)	1.43	5.27	−0.45
M_im (%)	−0.98	3.27	−2.29	M_im (%)	−0.48	3.27	−1.01

Table 9. Thermal Face-CNN accuracy, recall, and precision values 3.

Index	(10, 39, 33)			Index	(100, 39, 34)		
	Accuracy	Recall	Precision		Accuracy	Recall	Precision
Average	0.7845	0.7863	0.758	Average	0.7843	0.7535	0.7731
The best	0.8245	0.8938	0.8218	The best	0.8327	0.8319	0.8103
A_im (%)	0.37	12.39	−5.51	A_im (%)	0.34	7.15	−3.76
M_im (%)	−1.46	13.48	−3.05	M_im (%)	−0.48	5.33	−4.6

Table 10. Thermal Face-CNN accuracy, recall, and precision values 4.

Index	(−5, 39, 34)			Index	(−10, 39, 34)		
	Accuracy	Recall	Precision		Accuracy	Recall	Precision
Average	0.8151	0.7956	0.8027	Average	0.8033	0.7903	0.7853
The best	0.8367	0.8407	0.8515	The best	0.8367	0.8673	0.8214
A_im (%)	4.29	13.72	0.06	A_im (%)	2.7	11.47	−2.16
M_im (%)	0	6.74	0.46	M_im (%)	0	9.18	−3.19

Table 11. Thermal Face-CNN accuracy, recall, and precision values 5.

Index	(−100, 39, 34)			Index	(5, 39, 34)		
	Accuracy	Recall	Precision		Accuracy	Recall	Precision
Average	0.7912	0.7726	0.7755	Average	0.7939	0.7429	0.7972
The best	0.8163	0.8761	0.8367	The best	0.8367	0.8496	0.8349
A_im (%)	1.23	10.43	−3.33	A_im (%)	1.57	6.19	−0.63
M_im (%)	−2.43	11.24	−1.28	M_im (%)	0	7.87	−1.5

Table 12. Thermal Face-CNN accuracy, recall, and precision values 6.

Index	(1000, 39, 34)		
	Accuracy	Recall	Precision
Average	0.7294	0.6372	0.7399
The best	0.7918	0.7434	0.8298
A_im (%)	−7.16	−9.79	−8.42
M_im (%)	−5.67	−5.95	−2.15

“The best” and “Average” in Tables 7–12 mean the highest value and average value, respectively. In Tables 7–12, A_im (%) means how much the average value is improved in comparison with CNN, and M_im (%) means how much the maximum value is improved in comparison with CNN. For example, A_im (%) and M_im (%) are obtained by average and the best values in the right side of Tables 4 and 7, Tables 8–12. The information on all the experimental results can be found at websites listed in the Supplementary Materials found at the end of this paper.

When the *knowledge value* is 10 in the Thermal Face-CNNs described in Tables 7 and 8 and the left side of Table 9, the values of accuracy, recall, and precision are obtained as changes occur to the values of the *up limit* and *down limit*. When the *up limit* and *down limit* are 39 and 33, respectively, the average recall value has the greatest increase, by 12.39%. When the *up limit* and *down limit* values are 39 and 34, respectively, the average recall value is increased by 10.44%. When the *up limit* and *down limit* are 40 and 34, respectively, the average recall value is increased by 7.97%, and the average precision value is decreased slightly by −1.53%. In addition, when the *up limit* and *down limit* are 41 and 34, respectively, the average recall is increased by 6.61%, and the precision is decreased by −2.18%. When the values of the *up limit* and *down limit* are 39 and 35, respectively, the amount of the increment of recall is reduced the best.

The Thermal Face-CNNs described on the left side of Table 7 and the right side of Tables 9 and 10, Tables 11 and 12 show the amount by which the performance changed when the *up limit* and *down limit* are 39 and 34, respectively, and when the *knowledge value* is changed. Table 12 shows that much lower performance can be achieved with Thermal Face-CNN than with CNN. The Thermal Face-CNN used to obtain the data in Table 12 has the same parameters as the Thermal Face-CNNs used to obtain the data in the left side of Table 7, except for the fact that the *knowledge value* is 1,000. Therefore, a huge *knowledge value* shows that performance can be rather reduced. The best performance was obtained by increasing the average recall value by 13.72% when the *knowledge value* was −5, and the second-best average recall value was increased by 11.47% when the *knowledge value* was −10. In addition, when the *knowledge value* was 10, the third-best performance was obtained by increasing the average recall value by 10.44%. When the *knowledge value* was −100, the average recall value was increased by 10.43%, which was the fourth-best performance.

Except for Table 12, the average recall values of the Thermal Face-CNN having external knowledge about the temperature of the real face in Tables 7–11 show that the average recall value and the best recall value are better than the CNN shown in the right side of Table 4. An increase of the recall value means that the Thermal Face-CNN has detected more data having the real face than CNN. It can be seen that CNN and Thermal Face-CNN are not significantly different in terms of accuracy and precision when we compare the values in the right sides of Tables 4 and 7, Tables 8–11. Looking at

the performance of Thermal Face-CNN that obtained the best performance, in the left side of Table 10, we can see that Thermal Face-CNN was not reduced at all. Therefore, Thermal Face-CNN is superior to CNN in all indices.

The performance obtained by Thermal Face-CNN must be compared with the accuracy, recall, and precision values recorded in Tables 5 and 6 quantitatively. Table 10 shows that the method with the highest accuracy is 0.8367 on Thermal Face-CNN. In addition, the results in Table 6 show that C-SVM is the method with the highest recall. Further, Table 5 shows that MLP is the method with the highest precision. However, MLP is a relatively bad way to detect the real face because the recall value is too small. Thermal Face-CNN has the best accuracy and more balance between recall and precision than MLP and C-SVM. For accurate performance evaluation, F-measure [40] is used. F-measure is a widely used index that quantitatively evaluates performance by simultaneously considering recall and precision. F-measure is shown in Equation (2).

$$F\text{-measure} = \frac{(\beta^2 + 1) \times \text{precision} \times \text{recall}}{\beta^2 \times \text{precision} + \text{recall}} \quad (2)$$

β is a positive real number or zero. Also *precision*, *recall*, and *F-measure* are the values of precision, recall, and F-measure, respectively. A larger F-measure value means a better algorithm. When β is one, the most frequently used F-measure formula appears in Equation (3).

$$F\text{-measure}_1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (3)$$

*F-measure*₁ in Equation (3) means the value of F-measure when β is one. As shown in Equation (4), *difference* denotes the difference value of F-measures of the Thermal Face-CNN and C-SVM; Thermal Face-CNN obtained 0.8327 accuracy, 0.8407 recall, 0.8051 precision, and C-SVM obtained 0.8245 accuracy, 0.9381 recall, 0.7465 precision corresponding to Table 6.

$$\text{difference} = \frac{(\beta^2 + 1) \times 0.8051 \times 0.8407}{\beta^2 \times 0.8051 + 0.8407} - \frac{(\beta^2 + 1) \times 0.7465 \times 0.9381}{\beta^2 \times 0.7465 + 0.9381} \quad (4)$$

When the *difference* is zero, the β value is 0.8885, meaning that the two *f-measure* values are the same. When β is greater than or equal to 0 and less than 0.8885, then Thermal Face-CNN is better. By contrast, when β is greater than 0.8885, C-SVM is better. You can find the corresponding conditions by obtaining equations in the same way for several Thermal Face-CNNs. It is trivial to find β that makes *difference* zero when the parameters are different. Nevertheless, it is important to show that the Thermal Face-CNN is superior by listing the F-measures obtained at commonly used β values of 0.5 and 2. Table 13 shows it.

In Table 13, “Average F-measure” means the F-measure using average recall and average precision in the left side of Table 10. When β is 2, F-measure means that F-measure weighs recall higher than precision. When β is 0.5, F-measure means that F-measure weighs recall lower than precision. Therefore, we can see that Thermal Face-CNN is best when precision has more weight than recall. Precision is more important than recall when the reliability of the algorithm is important. Therefore, Thermal Face-CNN is good for this situation.

In addition to the comparison based on accuracy, recall, precision, and F-measure, it is shown that the CNN-based proposed algorithm is superior to CNN and has similar performance with the others on receiver operating characteristic (ROC) graph [41] in Figure 5. Parenthetical values in Figure 5 indicate *knowledge value*, *up limit*, and *down limit* values, sequentially.

Table 13. C-SVM's and Thermal Face-CNN's F-measure comparison.

c	Which algorithm is superior?			
	F-measure on C-SVM with RBF and $\beta = 0.5$	F-measure on C-SVM with RBF and $\beta = 2$	When Thermal Face-CNN has $(-5, 39, 34)$ and $\beta = 0.5$, Average F-measure = 0.8013	When Thermal Face-CNN has $(-5, 39, 34)$ and $\beta = 2$, Average F-measure = 0.797
0.7	0.0425	0.011	Thermal Face-CNN	Thermal Face-CNN
0.8	0.7692	0.8893	Thermal Face-CNN	C-SVM
0.81	0.7604	0.8863	Thermal Face-CNN	C-SVM
0.82	0.7686	0.9068	Thermal Face-CNN	C-SVM
0.83	0.7686	0.9068	Thermal Face-CNN	C-SVM
0.84	0.7602	0.9038	Thermal Face-CNN	C-SVM
0.85	0.7559	0.9023	Thermal Face-CNN	C-SVM
0.86	0.7559	0.9023	Thermal Face-CNN	C-SVM
0.87	0.7559	0.9023	Thermal Face-CNN	C-SVM
0.88	0.7559	0.9023	Thermal Face-CNN	C-SVM
0.89	0.7602	0.9038	Thermal Face-CNN	C-SVM
0.9	0.7602	0.9038	Thermal Face-CNN	C-SVM
0.91	0.7644	0.9053	Thermal Face-CNN	C-SVM
0.92	0.7686	0.9068	Thermal Face-CNN	C-SVM
0.93	0.7686	0.9068	Thermal Face-CNN	C-SVM
0.94	0.7686	0.9068	Thermal Face-CNN	C-SVM
0.95	0.7632	0.8932	Thermal Face-CNN	C-SVM
0.96	0.7648	0.8878	Thermal Face-CNN	C-SVM
0.97	0.7692	0.8893	Thermal Face-CNN	C-SVM
0.98	0.7692	0.8893	Thermal Face-CNN	C-SVM
0.99	0.7738	0.8908	Thermal Face-CNN	C-SVM
1	0.7783	0.8923	Thermal Face-CNN	C-SVM
1.5	0.7755	0.8853	Thermal Face-CNN	C-SVM

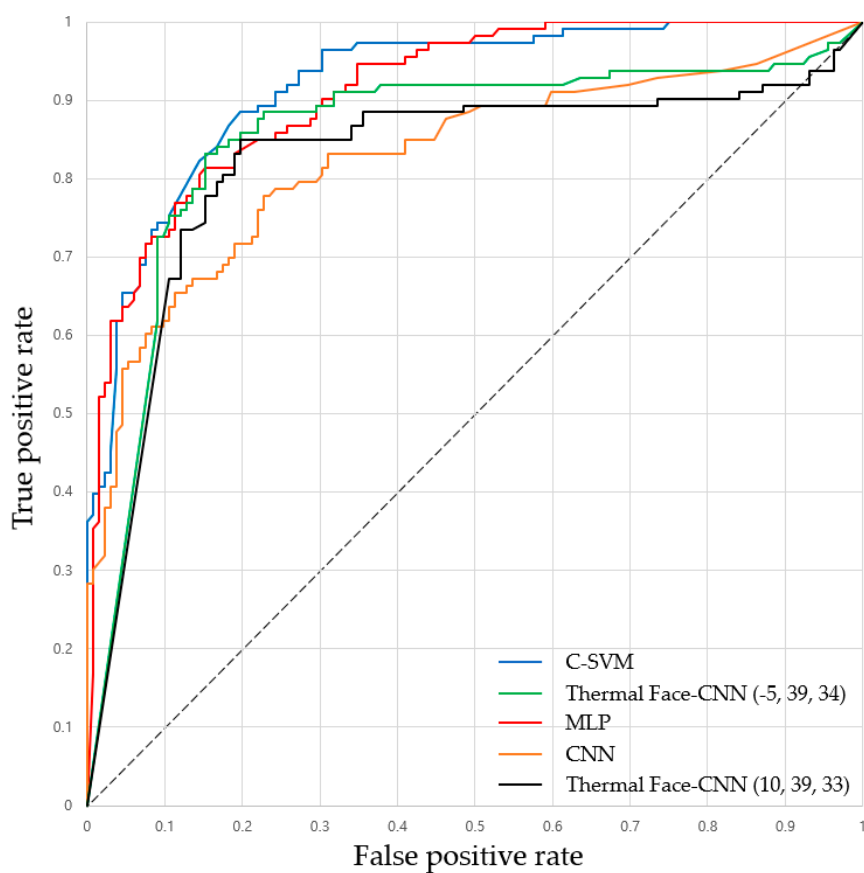


Figure 5. Receiver operating characteristic (ROC) graph.

'A' line is better than 'B' line if 'A' line is closer to the northwest than 'B' line in ROC graph. The blue line in Figure 5 shows the performance of C-SVM, the green and black lines show the performance of Thermal Face-CNN, the red line shows the performance of MLP, and the orange line shows the performance of CNN. To obtain Figure 5, we used the parameters having the best performance: MLP which has an accuracy of 0.7837, a recall of 0.5664, and a precision of 0.9412 and the CNN which has an accuracy of 0.8367, a recall of 0.7876, and a precision of 0.8476 and the best performance among a *up limit* value of 39, and a *down limit* value of 34 in Thermal Face-CNN which has an accuracy of 0.8327, a recall of 0.8407, a precision of 0.8051, a *knowledge value* value of -5 , a *up limit* value of 39, and a *down limit* value of 34 and the best performance among a *knowledge value* of 10 in Thermal Face-CNN which has an accuracy of 0.8245, a recall of 0.8496, a precision of 0.7869, a *knowledge value* value of 10, a *up limit* value of 39, and a *down limit* value of 33 and C-SVM which has a c value of 1 are used. As shown in Figure 5, Thermal Face-CNN has the dramatic performance improvement compared to CNN, and the Thermal Face-CNN's performance is close to that of MLP and C-SVM. In this paper, we argue that Thermal Face-CNN is better when precision is more important than recall. However, ROC graph does not directly consider precision because it uses true positive rate and false positive rate, which are not precision. Nonetheless, the ROC graph shows that Thermal Face-CNN is superior to CNN.

5. Conclusions and Future Works

Face liveness detection is an important field that allows for information about a real person to be communicated when communicating security. In this paper, face liveness detection was performed in indoor residential environment using the fact that thermal patterns on a face in a display and a photograph differ from those on the real face. First, we quantitatively compared the performance of the thermal image with the RGB image. It has been shown that the thermal image is more suitable for face liveness detection because CNN has the best performance, with an accuracy of 0.6898, a recall of 0.5752, a precision of 0.7342 on the RGB image dataset, and an accuracy of 0.8367, a recall of 0.7876, and a precision of 0.8476 on the thermal image dataset. We also propose Thermal Face-CNN, which has external knowledge about the real face temperature in the existing CNN algorithm and compares it with CNN. The performance of the best-performing Thermal Face-CNN is equal to or better than CNN. Furthermore, we used the F-measure to identify the condition in which the Thermal Face-CNN performs better than the C-SVM.

Based on the results in this paper, we hope that Thermal Face-CNN with the thermal image is used to detect malicious tricks to imitate the face. This paper shows that it is possible to insert external knowledge by adjusting the value of a particular real number range. Therefore, it is expected that the application algorithms that have knowledge in various fields will emerge.

In this study, the experiment was conducted using 844 scenes. Nevertheless, as the number of data increases, it becomes more feasible to use face liveness detection in more general situations. Therefore, there is a need to collect thermal images in the future. Moreover, due to the difference between the RGB lens and the infrared lens, the images measured differ in terms of pixel size, the number of pixels, and the range of the scene. Therefore, there is a need to construct datasets with fewer differences between the RGB and thermal image. Because the experiments of all the possible combinations of the parameters in the algorithms were not done, the comparisons are not conclusive. Therefore, it is necessary to accurately identify the optimal parameters combination that obtains the highest accuracy, recall, precision, F-measure value through additional experimentation.

Supplementary Materials: The information about Flir C3 is available online at <https://www.flir.com/products/c3/>. The RGB image dataset is available online at https://www.researchgate.net/publication/328297217_RGB_Face_Liveness_Detection_Dataset_JongwooRFLDDJ. The thermal image dataset is available online at https://www.researchgate.net/publication/327716173_Thermal_Face_Liveness_Detection_Dataset_JongwooTFLDDJ. SVC is available online at <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>. All the experimental results are available online at https://www.researchgate.net/publication/330359019_symmetry_experiments.

The raw images are available online at https://www.researchgate.net/publication/330382261_Raw_images_for_Face_Liveness_Detection_Using_Thermal_Face-CNN_with_External_Knowledge.

Author Contributions: Conceptualization, J.S.; data curation, J.S.; formal analysis, J.S.; investigation, J.S.; methodology, J.S.; project administration, I.-J.C.; resources, J.S.; software, J.S.; supervision, I.-J.C.; validation, I.-J.C. and J.S.; writing—original draft, J.S.; writing—review & editing, I.-J.C. and J.S.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Svozil, D.; Kvasnicka, V.; Pospichal, J. Introduction to multi-layer feed-forward neural networks. *Chemom. Intell. Lab. Syst.* **1997**, *39*, 43–62. [[CrossRef](#)]
2. Lawrence, S.; Giles, C.L.; Tsoi, A.C.; Back, A.D. Face recognition: A convolutional neural-network approach. *IEEE Trans. Neural Netw.* **1997**, *8*, 98–113. [[CrossRef](#)] [[PubMed](#)]
3. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
4. Powers, D.M. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–63.
5. Jiang, H.; Learned-Miller, E. Face detection with the faster R-CNN. In Proceedings of the 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), Washington, DC, USA, 30 May–3 June 2017; pp. 650–657.
6. Patel, K.; Han, H.; Jain, A.K. Secure face unlock: Spoof detection on smartphones. *IEEE Trans. Inf. Forensics Secur.* **2016**, *11*, 2268–2283. [[CrossRef](#)]
7. Wu, L.; Xu, Y.; Xu, X.; Qi, W.; Jian, M. A face liveness detection scheme to combining static and dynamic features. In Proceedings of the Chinese Conference on Biometric Recognition, Chengdu, China, 14–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 628–636.
8. Tan, X.; Li, Y.; Liu, J.; Jiang, L. Face liveness detection from a single image with sparse low rank bilinear discriminative model. In Proceedings of the European Conference on Computer Vision, Crete, Greece, 5–11 September 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 504–517.
9. Kim, G.; Eum, S.; Suhr, J.K.; Kim, D.I.; Park, K.R.; Kim, J. Face liveness detection based on texture and frequency analyses. In Proceedings of the 2012 5th IAPR International Conference on Biometrics (ICB), New Delhi, India, 29 March–1 April 2012; pp. 67–72.
10. Chingovska, I.; Anjos, A.; Marcel, S. On the effectiveness of local binary patterns in face anti-spoofing. In Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG), Darmstadt, Germany, 6–7 September 2012.
11. Zhang, Z.; Yi, D.; Lei, Z.; Li, S.Z. Face liveness detection by learning multispectral reflectance distributions. In Proceedings of the 2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011), Santa Barbara, CA, USA, 21–25 March 2011; pp. 436–441.
12. Erdogmus, N.; Marcel, S. Spoofing face recognition with 3D masks. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 1084–1097. [[CrossRef](#)]
13. Pan, G.; Sun, L.; Wu, Z.; Lao, S. Eyeblink-Based Anti-Spoofing in Face Recognition from a Generic Webcam. In Proceedings of the 2007 11th IEEE International Conference on Computer Vision (ICCV), Rio de Janeiro, Brazil, 14–20 October 2007; pp. 1–8.
14. Anjos, A.; Marcel, S. Counter-measures to photo attacks in face recognition: A public database and a baseline. In Proceedings of the 2011 International Joint Conference on Biometrics (IJCB), Washington, DC, USA, 11–13 October 2011; pp. 1–7.
15. Zhang, Z.; Yan, J.; Liu, S.; Lei, Z.; Yi, D.; Li, S.Z. A face antispoofing database with diverse attacks. In Proceedings of the 2012 5th IAPR International Conference on Biometrics (ICB), New Delhi, India, 29 March–1 April 2012; pp. 26–31.
16. Wen, D.; Han, H.; Jain, A.K. Face spoof detection with image distortion analysis. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 746–761. [[CrossRef](#)]
17. Patel, K.; Han, H.; Jain, A.K.; Ott, G. Live face video vs. spoof face video: Use of moiré patterns to detect replay video attacks. In Proceedings of the 2015 International Conference on Biometrics (ICB), Phuket, Thailand, 19–22 May 2015; pp. 98–105.

18. Pinto, A.; Schwartz, W.R.; Pedrini, H.; Rocha, A.D. Using visual rhythms for detecting video-based facial spoof attacks. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 1025–1038. [[CrossRef](#)]
19. Pinto, A.; Pedrini, H.; Schwartz, W.R.; Rocha, A. Face spoofing detection through visual codebooks of spectral temporal cubes. *IEEE Trans. Image Process.* **2015**, *24*, 4726–4740. [[CrossRef](#)] [[PubMed](#)]
20. König, D.; Adam, M.; Jarvers, C.; Layher, G.; Neumann, H.; Teutsch, M. Fully convolutional region proposal networks for multispectral person detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017; pp. 243–250.
21. Zhang, X.; Chen, G.; Saruta, K.; Terata, Y. Deep Convolutional Neural Networks for All-Day Pedestrian Detection. In *Information Science and Applications*; Springer: Singapore, 2017; pp. 171–178.
22. Baek, J.; Hong, S.; Kim, J.; Kim, E. Efficient pedestrian detection at nighttime using a thermal camera. *Sensors* **2017**, *17*, 1850. [[CrossRef](#)] [[PubMed](#)]
23. Kwaśniewska, A.; Rumiński, J. Face detection in image sequences using a portable thermal camera. In Proceedings of the 13th Quantitative Infrared Thermography Conference, Quebec City, QC, Canada, 4–8 July 2016.
24. Burges, C.J. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. [[CrossRef](#)]
25. Peña, J.; Gutiérrez, P.; Hervás-Martínez, C.; Six, J.; Plant, R.; López-Granados, F. Object-based image classification of summer crops with machine learning methods. *Remote Sens.* **2014**, *6*, 5019–5041. [[CrossRef](#)]
26. Alotaibi, A.; Mahmood, A. Deep face liveness detection based on nonlinear diffusion using convolution neural network. *Signal Image Video Process.* **2017**, *11*, 713–720. [[CrossRef](#)]
27. Akbulut, Y.; Şengür, A.; Budak, Ü.; Ekici, S. Deep learning based face liveness detection in videos. In Proceedings of the 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 16–17 September 2017; pp. 1–4.
28. Peixoto, B.; Michelassi, C.; Rocha, A. Face liveness detection under bad illumination conditions. In Proceedings of the 2011 18th IEEE International Conference on Image Processing (ICIP), Brussels, Belgium, 11–14 September 2011; pp. 3557–3560.
29. Boulkenafet, Z.; Komulainen, J.; Hadid, A. Face spoofing detection using colour texture analysis. *IEEE Trans. Inf. Forensics Secur.* **2016**, *11*, 1818–1830. [[CrossRef](#)]
30. Sund-Levander, M.; Forsberg, C.; Wahren, L.K. Normal oral, rectal, tympanic and axillary body temperature in adult men and women: A systematic literature review. *Scand. J. Caring Sci.* **2002**, *16*, 122–128. [[CrossRef](#)] [[PubMed](#)]
31. Cong, J.; Xiao, B. Minimizing computation in convolutional neural networks. In Proceedings of the International Conference on Artificial Neural Networks, Hamburg, Germany, 15–19 September 2014; Springer: Cham, Switzerland, 2014; pp. 281–290.
32. Scherer, D.; Müller, A.; Behnke, S. Evaluation of pooling operations in convolutional architectures for object recognition. In *Artificial Neural Networks—ICANN 2010*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 92–101.
33. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
34. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
35. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
36. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
37. Jain, A.K.; Mao, J.; Mohiuddin, K.M. Artificial neural networks: A tutorial. *Computer* **1996**, *29*, 31–44. [[CrossRef](#)]
38. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, Savannah, GA, USA, 2–4 November 2016; Volume 16, pp. 265–283.
39. Chang, Y.-W.; Hsieh, C.-J.; Chang, K.-W.; Ringgaard, M.; Lin, C.-J. Training and testing low-degree polynomial data mappings via linear SVM. *J. Mach. Learn. Res.* **2010**, *11*, 1471–1490.

40. Sokolova, M.; Japkowicz, N.; Szpakowicz, S. Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation. In Proceedings of the Australasian Joint Conference on Artificial Intelligence, Hobart, Australia, 4–8 December 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 1015–1021.
41. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).