Technical Note

Heart ID: Human Identification Based on Radar Micro-Doppler Signatures of the Heart Using Deep Learning

Peibei Cao, Weijie Xia * and Yi Li

Key Laboratory of Radar Imaging and Microwave Photonics (Nanjing Univ. Aeronaut. Astronaut.), Ministry of Education, College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, 29 Jiangjun Street, Nanjing 211100, China; caopeibei@nuaa.edu.cn (P.C.);
liyi1995@nuaa.edu.cn (Y.L.)

* Correspondence: nuaaxwj@nuaa.edu.cn; Tel.: +86-139-1471-8730

Received: 1 April 2019; Accepted: 16 May 2019; Published: 23 May 2019

Abstract: Human identification based on radar signatures of individual heartbeats is crucial in various applications, including user authentication in mobile devices, identification of escaped criminals, etc. Usually, optical systems employed to recognize humans are sensitive to ambient light environments, while radar does not have such a drawback, since it has high penetration and all-weather capability. Meanwhile, since micro-Doppler characteristics from the heart of different people are distinct and not easy to fake, it can be used for identification. In this paper, we employed a deep convolutional neural network (DCNN) and conventional supervised learning methods to realize heartbeat-based identification. First, the heartbeat signals were acquired by a Doppler radar and processed by short-time Fourier transform. Then, predefined features were extracted for the conventional supervised learning algorithms, while time–frequency graphs were directly inputted to the DCNN since the network had its own feature extraction part. It is shown that the DCNN could achieve average accuracy of 98.5% for identifying four people, and higher than 80% when the number of people was less than ten. For conventional supervised learning algorithms when identifying four people, the accuracy of the support vector machine (SVM) was 88.75%, and the accuracy of SVM–Bayes was 91.25%, while naive Bayes had the lowest accuracy of 80.75%.

Keywords: human identification; micro-Doppler; heartbeat; DCNN; SVM; NB; SVM–Bayes fusion

1. Introduction

Human identification has various applications, especially in security and surveillance. Hence, it is drawing increasing attention [1–3]. In existing works, human identification is usually based on pictures or videos, which are susceptible to light and will invade people’s privacy to some extent, limiting their use under many circumstances. Compared with the optical surveillance system, Doppler radar can be used in no light and bad weather conditions, making it a great tool to detect and classify targets without violating users’ privacy. However, the radar also has some drawbacks like radiation [4]. In addition to the main Doppler shift, the micro-Doppler signal is caused by the constant or slow motion of the target, such as the wheels of vehicles, the heartbeat of a human, etc. The micro-Doppler signature has been investigated for various applications including activity classification, battlefield monitoring, and so on [5–7]. Furthermore, heartbeats extracted from radio frequency (RF) signals have been investigated in many studies. In [8], heartbeats were employed for emotion classification, while in [9], heartbeats were used for authentication. Human identification based on micro-Doppler signatures from the heart is still in current research.
In this paper, a deep learning method together with conventional supervised learning methods and fusion algorithms of conventional supervised learning methods are employed to deal with human identification problems based on micro-Doppler signatures from the heart. In prior works, the performance of a deep convolutional neural network (DCNN) is usually better than conventional supervised learning methods like pattern recognition, image recognition, and speech recognition [10–12] without any feature extraction process, since the DCNN can jointly learn features and classification boundaries directly from raw input data. Therefore, we expect to yield good results for the human identification problem by exploiting a DCNN. We will present our experimental results and brief backgrounds of the support vector machine (SVM) and naive Bayes (NB). Additionally, the fusion algorithm of the support vector machine and Bayes (SVM–Bayes) is carried out for a better result of human identification based on micro-Doppler characteristics from the heart.

The remaining paper is organized as follows. Section 2 illustrates experiment setup and data processing. Brief backgrounds on DCNNs, SVM, NB and the SVM–Bayes fusion algorithm are described in Section 3. Section 4 presents the experiment results and discusses the impact of noise and number of humans for the four algorithms. Section 5 concludes the paper.

2. Experiment Setup and Data Analysis

2.1. Experiment Setup

The experiments were performed in the laboratory of the college of Electronic and Information Engineering in Nanjing University of Aeronautics and Astronautics, which is shown in Figure 1a. Figure 1b shows the equipment and system deployment of our experiments. The system, which was employed to collect experimental micro-Doppler signatures of targets, included an IVS-179 radar, an M2i.4912 eight-channel parallel data acquisition card, and an ACME industrial personal portable computer. The IVS-179 radar worked at 24 GHz in the continuous wave (CW) mode without modulation and the ACME computer recorded industrial samples at 50 kHz.

![Figure 1. Data collection setup and experiment scenario. (a) Experiment scenario; (b) The equipment and system deployment of our experiments.](image)

We collected data from 10 people, half of whom were women and the other half were men. Each person sat about 1.5 m in front of the radar for about 6 seconds without breathing. For each person, we repeated this procedure 100 times.

2.2. Micro-Doppler Signatures and Choice of Features

According to the Doppler effect, when a target has motion relative to the wave source, there will be a change in the wavelength of a wave. Meanwhile, if some parts of the target have micro-motion, there will be the micro-Doppler effect, where the additional frequency components in addition to the main Doppler shift can be observed in the joint time–frequency space [13]. Therefore, short-time Fourier transform (STFT) is exploited to characterize micro-Doppler signatures.
Suppose the window function is $g(t)$, which will slide along the time line. The time domain Doppler signal is $x(t)$, and its STFT [14] can be expressed as

$$\mathcal{F}_{STFT}(t, f) = \int_{-\infty}^{+\infty} x(u) g^*(u-t) \exp(-j2\pi fu) du.$$  \hfill (1)

In our work, we chose a Gaussian window, so $\mathcal{F}_{STFT}(t, f)$ can be expressed as

$$\mathcal{F}_{STFT}(t, f) = \int_{-\infty}^{+\infty} x(u) \exp\left(-\frac{(t-u)^2}{2\sigma^2}\right) \exp(-j2\pi fu) du,$$  \hfill (2)

In the process, the proper time window size and sliding step are vital in capturing the micro-Doppler feature of a target in the time–frequency domain. After repeated practice, we chose 0.139 s as the time window and set the sliding step size to 1/2000 s. The spectrograms of heartbeats from four different people are shown in Figure 2. It is apparent that the heartbeat for everyone is unique. First, from the graphs, we can see that the heartbeat differs in period (horizontal distance between yellow/red dots) and energy (color of the dots). Additionally, their bandwidth (vertical extension) and shape are different from each other. In order to get sufficient data for the DCNN, we divided the test data into three parts equally. After that, we employed basic image transformation methods to the graphs like translation, rotation, zoom, mirroring, and cropping. These methods are often used to augment the dataset, as the character of the images will not be altered by these transformations, which means the classification results will not be influenced. After these operations, the number of spectrograms for each person was 4000.

**Figure 2.** Time–frequency graphs for four different people. (a) Time–frequency graph for person 1; (b) Time–frequency graph for person 2; (c) Time–frequency graph for person 3; (d) Time–frequency graph for person 4.
For conventional supervised learning methods, we extracted features from the spectrograms. Features adopted in this paper are listed as follows:

1. The period of the heartbeat;
2. The energy of the heartbeat;
3. The bandwidth of the Doppler signal.

The total energy of the radar signal is

\[ E = \int S_B(t) \ast S^*_B(t) \]  

where \( E \) represents the energy of the signal and \( S_B \) represents the radar signal.

We normalize the energy by dividing each signal through its respective energy:

\[ E_{\text{norm}} = \frac{S_B(t)}{E} \]  

where \( E_{\text{norm}} \) represents normalized energy.

2.3. Experiment on the Penetrability of the Radar

The radar we employed was a 24 GHz radar, with which we did experiments on the radar penetration of skin. The experiment was performed in the laboratory of the college of Electronic and Information Engineering in Nanjing University of Aeronautics and Astronautics, which is shown in Figure 3. During the experiment, a steel plate and pieces of pork were used to simulate a moving target. They moved back and forth as a rope was pulled. We used two pieces of pork to replace human tissue for the experiment (see Figure 4). The thicknesses of the pieces of pork were 9 mm and 13 mm (+/- 2 mm measurement error). The pork was placed between the radar and the pendulous steel plate. When there was no pork between the radar and the pendulous steel plate, the radar could receive the reflection from the steel plate, which is shown in Figure 5b. When there was no subject in front of the radar, no micro-Doppler signal could be observed (Figure 5a). When pork A was placed between the radar and pendulous pork B, no micro-Doppler signal could be observed (Figure 5e). Therefore, a steel plate was a valid analogue for a human heart. We performed the experiment with pork A only (Figure 5c), and pork A and B (Figure 5d) between the radar and the pendulous steel plate. We observed that the micro-Doppler effect of the signal was obvious, which indicated that the shielding effect of pork A was weak and most of the signal could penetrate it. If pork A and B were between the radar and the pendulous steel plate, we observed that the micro-Doppler effect was obviously weakened, which indicated that the shielding effect became stronger. In conclusion, with an increase in thickness, less signal can be received by the radar. Considering the thickness of people’s thoraxes, the signal we received was from the heart.

**Figure 3.** Setup of the experiment on the penetrability of the radar.
Figure 4. The pork we employed in the experiment.

(a)                                          (b) 

(c)                                          (d)

Remote Sens. 2019, 11, x FOR PEER REVIEW 5 of 16

Figure 5. Cont.
A deep convolutional neural network (DCNN) is a successful deep learning algorithm that can be employed to classify new samples or intelligently predict possibilities by learning the underlying patterns and characteristics from existing data. In recent years, it has performed well in various fields such as image recognition and speech recognition in numerous researchers’ studies. It is a kind of multilayer supervising learning neural network. The key components of a DCNN are the convolution and pooling operation in the hidden layer, which are used for feature extraction. Multiple convolution filters work in parallel on input data to build the feature maps in the convolutional layer followed by the pooling layer. A simple DCNN architecture with two convolution layers, two pooling layers, and one fully connected layer is shown in Figure 6. Figure 7 shows the schematic diagram of the convolutional and pooling operation. The layers after the feature extraction part of the DCNN are full connection layers, which includes a logistic regression classifier. The input of the full connection is the output of the last pooling layer, and the output of the full connection is the classification results. In our work, we chose softmax regression to classify characteristics. Softmax is developed from logistic regression in order to solve multiclass problems. The function of softmax can be expressed as

$$h_{\theta}(x) = \begin{bmatrix} p(y^{(i)} = 1|x^{(i)}; \theta) \\ p(y^{(i)} = 2|x^{(i)}; \theta) \\ \vdots \\ p(y^{(i)} = k|x^{(i)}; \theta) \end{bmatrix},$$

where $p(y^{(i)} = j|x^{(i)}; \theta)$ represents the probability of the input $x^{(i)}$ of the $i$th sample belonging to category $j$. $\theta$ represents the parameters of the training model. The loss function of the softmax classifier can be expressed as

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{k} 1[y_i = j] \log \frac{\exp(\theta_j^T x_i)}{\sum_{l=1}^{k} \exp(\theta_l^T x_i)}.$$  

In this work, stochastic gradient descent (SGD) was used to minimize the loss function $J(\theta)$ using the back propagation algorithm until the network converged or reached the maximum iteration. To effectively prevent overfitting, AlexNet employed a dropout operation [11]. The neurons that are in the state of dropout do not participate in the forward propagation or in the back propagation.
In this way, the neural network tries a new structure for every input sample, reducing the complex interrelationship of neurons.

![A simple deep convolutional neural network (DCNN) architecture.](image)

**Figure 6.** A simple deep convolutional neural network (DCNN) architecture.

![Process of applying a 4 × 4 convolution filter to the input data to generate the output (in gray). Examples of 2 × 2 pooling (max or mean pooling).](image)

**Figure 7.** (a) Process of applying a 4 × 4 convolution filter to the input data to generate the output (in gray). (b) Examples of 2 × 2 pooling (max or mean pooling).

3.2. **Support Vector Machine (SVM)**

The support vector machine (SVM) based on statistical learning theory (SLT) [15] has performed well in many areas. It is a binary classifier that defines the maximal margin hyperplane in the feature space, followed by the convex quadratic programming optimization algorithm [16]. When classifying targets, the hyperplane is utilized to separate a given set of binary labeled training data. In the conditions where the extracted features cannot be separated linearly, the technique of “kernels” is used to realize a non-linear mapping to a feature space where the linear separation exists.

The $j$th input point $x^j = (x_1^j, \ldots, x_n^j)$, labeled by the variable $y^j \in \{-1, 1\}$, is the realization of the random vector $X^j$. $\phi(x)$ is the eigenvector of $x$ after the mapping with the technique of “kernels”. In this paper, we chose Gaussian kernels: $K(x^i, x^j) = \exp \left( -\frac{|x^i - x^j|}{\sigma^2} \right)$, where $\sigma$ is the kernel parameter.

In the feature space, the corresponding decision function is

$$f(x) = w^T \phi(x) + b,$$

where $w$ is the maximal margin hyperplane:

$$w = \sum_{i=1}^{m} \alpha_i y^i \phi(x^i),$$

while $\alpha_i$ are positive real numbers that maximize

$$\sum_{i=1}^{m} \alpha_i - \sum_{i,j=1}^{m} \alpha_i \alpha_j y^i y^j (\phi(x^i), \phi(x^j))$$

subject to

$$\sum_{i=1}^{m} \alpha_i y^i = 0, \quad \alpha_i > 0.$$  

The decision function can be equivalently expressed as
\[
f(x) = \sum_{i=1}^{m} \alpha_i y_i \phi(x')^T \phi(x) + b. \tag{11}
\]

SVM is a two-class classifier. In order to apply it to the multiclass problem like the one at hand, it should be modified into several binary class problems. Several works have been done to handle multiclass problems using SVMs [17,18]. In this paper, we employed one of these methods that combines SVMs with a decision-tree classifier. The configuration of this method is shown in Figure 8. The decision-tree method breaks multiple classes into several distinct binary decision problems using a tree structure. At each nonleaf node of the structure, a binary SVM classifier is used, which is simple and intuitive.

![Decision-tree based support vector machine (SVM) classifier.](image)

**Figure 8.** Decision-tree based support vector machine (SVM) classifier.

The complexity of SVM is not decided by the number of samples of the training set, but is related to the number of the supported vectors, which indicates that SVM has a simple system structure. Furthermore, it costs little time in the training and testing period, making it popular for pattern recognition, image recognition, and many other fields.

3.3. Naive Bayes (NB)

Naive Bayes (NB), which is based on Bayes’ [19,20] theorem, has the smallest misclassification rate when the conditional independence assumption is valid. Theoretically, the assumption will limit its application, but in fact, not only does the complexity of naive Bayes reduce exponentially, but also considerable robustness and efficiency are shown in many conditions that are opposition to the assumption. Considering its efficient evaluation, high accuracy, and solid theory foundation, it has been successfully applied to data mining tasks including classification, clustering, and the selection of models. Theoretically, the probability model for the Naive Bayes classifier can be expressed as follows.

For a given dataset \( (T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}) \), the Bayes formula can be expressed as

\[
P(Y | X) = \frac{P(X | Y) P(Y)}{\sum_k [P(X | Y_k) P(Y_k)]}.
\]  
(12)

Naive Bayes learns the joint probability \( P(X, Y) \) of input and output based on the conditional independence assumption. On this basis, for a given input \( X \), the method figures out the output \( Y \) with the maximum posterior probability using Bayes’ theorem. Naive Bayes can be equivalently expressed as

\[
y = f(x) = \arg\max_{\alpha_k} \frac{P(Y = \alpha_k) \prod_j P(X^{(j)} = x^{(j)} | Y = \alpha_k)}{\sum_k P(Y = \alpha_k) \prod_j P(X^{(j)} = x^{(j)} | Y = \alpha_k)}. \tag{13}
\]

When the input is \( x \), the category with the highest conditional probability is chosen as the class to be classified after calculating the conditional probability of all categories. Since the denominator of Formula (13) is the same for all \( \alpha_k \), Formula (13) can be simplified as
\[ y = f(x) = \arg\max_{c_k} P(Y = c_k) \prod_i P(X^{(i)} = x^{(i)} | Y = c_k). \quad (14) \]

### 3.4. SVM–Bayes Fusion

To explore whether better results can be achieved by the fusion of different conventional supervised learning methods, a fusion algorithm of SVM and Bayes was employed to process the extracted features. Figure 9 shows the schematic diagram of this fusion algorithm. In the process of Bayesian inference, when there is no empirical data available, the subjective probability can be used to substitute the prior probability as well as the likelihood function of the hypothetical event.

Let us assume that there are m kinds of feature extraction methods, n categories, and let Θ represent the collections of the categories (Θ = \{O_1, O_2, \cdots, O_n\}). The fusion algorithm is based on maximum a posteriori (MAP). For an unknown sample (O ∈ Θ), it can be expressed as

\[ P(O_{MAP}|O^1, O^2, \cdots, O^m) = \max_{O \in \Theta} P(O|O^1, O^2, \cdots, O^m), \quad (15) \]

where \( O^i \) is the decision for an unknown sample of SVM using the \( i \)th feature extraction method. \( O_{MAP} \) is the decision of the sample based on multi-feature extraction methods. \( P(O_{MAP}|O^1, O^2, \cdots, O^m) \) is the joint probability density function of SVM based on m kinds of feature extraction methods. Then Formula (15) can be expressed as

\[ O_{MAP} \triangleq \arg\max_{O \in \Theta} P(O|O^1, O^2, \cdots, O^m). \quad (16) \]

According to Bayes’ Formula (12), the fusion algorithm can be expressed as

\[ O_{MAP} \triangleq \arg\max_{O \in \Theta} \left\{ \frac{P(O^1, O^2, \cdots, O^m|O)P(O)}{\sum_{O \in \Theta} P(O^1, O^2, \cdots, O^m|O)P(O)} \right\}, \quad (17) \]

where \( P(O) \) is prior probability of \( O \in \Theta \).

The denominator of the equation is the full probability formula. This is not affected by the value of \( O \), so it can be simplified as

\[ O_{MAP} \triangleq \arg\max_{O \in \Theta} \left\{ P(O^1, O^2, \cdots, O^m|O)P(O) \right\}. \quad (18) \]

Since the feature extraction methods are independent, the equation can be expressed as

\[ O_{MAP} \triangleq \arg\max_{O \in \Theta} \left\{ P(O) \prod_{i=1}^{m} P(O^i|O) \right\}, \quad (19) \]

where \( P(O^i|O) \) is the likelihood function of SVM based on the \( i \)th feature extraction method. As the logarithm of this function is monotone increasing, the logarithm on the right side of Formula (19) can simplify the operation without affecting the fusion recognition decision. Then the equation can be expressed as

\[ O_{MAP} \triangleq \arg\max_{O \in \Theta} \left\{ \ln P(O) + \prod_{i=1}^{m} \ln P(O^i|O) \right\}. \quad (20) \]

\( P(O) \) and \( P(O^i|O) \) need to be confirmed. \( P(O) \) is prior probability, and in this case, it is assumed that each person has the same probability. The likelihood function \( P(O^i|O) \) means that when the identity of the person is certain, the probability that the person can be correctly identified by SVM. Its value can be approximated by the recognition experiment of training samples.
was adjusted to 0.001 because when the learning rate was larger, the loss would not converge, and when were followed by max pooling in each layer. We fine-tuned the parameters of the network according to with 4096 hidden nodes in the first fully connected layer, and an output layer as shown in Figure 10a.

The batch size determines the direction of gradient descent and the e

The size of the input pictures of the whole AlexNet must be 256

In this file, the spectrograms were resized to 227

For training data, 80% of the spectrograms of each person were used, and the rest were used to test. We employed Caffe [21], i.e., convolutional architecture, for fast feature embedding, which is open-source and speeded up by the NVIDIA Graphics Processing Unit (GPU) and Compute Unified Device Architecture (CUDA) library, as the platform to analyze the spectrograms. The GPU we used is the NVIDIA Quadro M4000.

We adopted AlexNet, in which there were five convolution layers, two fully connected layers with 4096 hidden nodes in the first fully connected layer, and an output layer as shown in Figure 10a. The size of the input pictures of the whole AlexNet must be 256 × 256, which can be normalized by the tools of Caffe directly. The partial internal structure of the network is shown in Figure 10. Figure 10b shows the first two convolutional layers and pooling layers, while Figure 10c shows the first full connection layer. In our work, rectified linear units (ReLU) were used as the activation function and were followed by max pooling in each layer. We fine-tuned the parameters of the network according to our experiments, and this was recorded in the configuration file of Caffe. The data of the learning rate was adjusted to 0.001 because when the learning rate was larger, the loss would not converge, and when the learning rate was smaller, the process was time-consuming. In our experiments, we allowed up to 5000 iterations for stable results. The learning rate was reduced to 0.001 × 0.9(floor(5000/2000)) after every 2000th iteration, since we employed the gradient descent method to solve the optimization problem. The weight decay was changed to 0.001 to prevent overfitting. The definition of the network for training and validation as well as the parameters for every layer were recorded in a particular file. In this file, the spectrograms were resized to 227 × 227, since the size of the input spectrograms of the first convolution layer must be 227 × 227. The batch sizes in the “train” part and “test” part were adjusted to 48 and 24 respectively. Batch size represents the number of samples taken in each iteration. The average of the gradient of those samples was used to update the parameters of the network. The batch size determines the direction of gradient descent and the effect and rate of convergence, as well as memory utilization. In the fully connected layer, we adapted the learning rate of bias to 15 and the learning rate of weight to 20 to speed up the learning rate in these layers. After 5000 iterations, our own network was generated and stored all relevant model parameters, like weight, that were suitable for our own spectrograms.

We assessed the performance of our network under the scenario where we recognized one person from a group. We randomly selected 100 spectrograms for each person from the test data to verify the stability of our own network, and calculated the time it took to classify one person. The whole process was repeated 100 times. The variance of the 100 times was 0.007, which means that the repeated 100 times varied slightly with each other, and our network was very stable.

![Figure 9. The framework of the SVM–Bayes fusion algorithm.](image_url)
which means that the process was not overfit. Figure 12 shows the confusion matrices of a random process for four humans. The loss shown is the value of the loss function in Formula (6). It can be observed that as the iteration grew, the accuracy and loss became stable, and the loss tended to 0, which means that the process was not overfit.

Figure 12 shows the confusion matrices of a random process for four humans. The loss shown is the value of the loss function in Formula (6). It can be observed that as the iteration grew, the accuracy and loss became stable, and the loss tended to 0, which means that the process was not overfit. Figure 12 shows the confusion matrices of a random process for four humans. The loss shown is the value of the loss function in Formula (6). It can be observed that as the iteration grew, the accuracy and loss became stable, and the loss tended to 0, which means that the process was not overfit.

Figure 11 shows the accuracy as well as the loss (see Equation (6)) of the training and testing process for four humans. The loss shown is the value of the loss function in Formula (6). It can be observed that as the iteration grew, the accuracy and loss became stable, and the loss tended to 0, which means that the process was not overfit. Figure 12 shows the confusion matrices of a random time for the identification results of the DCNN. Since the results of the repeated 100 times varied slightly, the result of one time could represent the accuracy to some extent. From the figures, we can see that as the number of the group became larger, the accuracy of the human identification declined. However, the average accuracy of the group below 10 was higher than 80%, which can be compared to the identification methods based on optical systems [22].

Figure 11. The accuracy and loss of the classification process for four humans.
The average accuracy of SVM is higher than that of NB, which is 88.75% and 80.75% respectively. The confusion matrices of SVM–Bayes, SVM, and NB respectively. It can be observed that SVM–Bayes has the highest accuracy—91.25% among the three algorithms, followed by SVM, and the lowest accuracy is NB. Though the results for each of them are satisfying, they all have high misclassification in identifying some people compared with the DCNN.

4.2. Based on Conventional Supervised Learning Algorithms

SVM and NB were applied to the extracted features (heartbeat period, energy, and bandwidth) for the human identification problem. The amount of data required in this part was relatively small compared with that utilized in the DCNN, which is one of the advantages of conventional supervised learning methods. In this paper, the fusion of two conventional supervised learning methods, SVM and Bayes, was employed with the aim of getting a better result. Figure 13 shows the confusion matrices of SVM–Bayes, SVM, and NB respectively. It can be observed that SVM–Bayes has the highest accuracy—91.25% among the three algorithms, followed by SVM, and the lowest accuracy is NB. The average accuracy of SVM is higher than that of NB, which is 88.75% and 80.75% respectively. Though the results for each of them are satisfying, they all have high misclassification in identifying some people compared with the DCNN.

Figure 12. The confusion matrices for the DCNN. (a) is four people, (b) is six people, (c) is eight people, and (d) is 10 people. The x and y axes indicate the serial number of the subjects.

Figure 13. Cont.
Figure 13. (a) Confusion matrix for SVM–Bayes fusion, (b) confusion matrix for SVM, (c) confusion matrix for naive Bayes (NB). The x and y axes indicate the serial number of the subjects.

4.3. The Impact of Noise and Human Number for the Four Algorithms

To study the robustness of our classification algorithms, we challenged each algorithm by adding noise to the respective input. We added different levels ($SNR = 29 \text{ dB} \sim 30 \text{ dB}, 20 \text{ dB} \sim 21 \text{ dB}, 10 \text{ dB} \sim 11 \text{ dB}, 0 \text{ dB} \sim 1 \text{ dB}$) of random noise to the raw radar signal. For example, we added random noise ($SNR = 0 \text{ dB} \sim 1 \text{ dB}$) to the raw signals that we collected in Section 2. After that, we randomly chose 100 spectrograms of each target at each noise grade to test the noise immunity of the algorithm. For each noise grade, the whole process was repeated 100 times, and the average accuracy was taken as the final result. As for detecting the anti-noise performance of SVM, NB, and the SVM–Bayes fusion algorithm, we also added different grades ($SNR = 29 \text{ dB} \sim 30 \text{ dB}, 20 \text{ dB} \sim 21 \text{ dB}, 10 \text{ dB} \sim 11 \text{ dB}, 0 \text{ dB} \sim 1 \text{ dB}$) of random noise to the signals that we collected in Section 2, and we followed this with feature extraction. Figure 14 compares the accuracy of the four different algorithms with different levels of noise. From this figure, we can observe that the DCNN had the highest accuracy of human identification at all noise levels, followed by SVM–NB, then SVM, and NB had the lowest accuracy. Additionally, it can be observed that the noise resistance of SVM–Bayes and SVM were very similar, while the DCNN had the best noise resistance and NB had the worst.

Figure 14. The impact of noise on the four classification algorithms.

Figure 15 shows the impact of the number of individuals in each test run of the four classification algorithms on the accuracy of correct identification. As can be observed, the accuracy of recognition decreased as the number of humans increased. Additionally, the DCNN had the best performance of the four algorithms. The accuracy of the SVM–Bayes fusion algorithm was a little bit higher than that of SVM, but as the number of people increased, the accuracy of the SVM–Bayes fusion algorithm dropped more quickly than SVM.
were far from satisfactory in both aspects. We believe that it shows the potential of DCNNs for human identification with a maximum group size of 5 [24]. The accuracy for a group of 2 people and 77% for a group of 6 people, which shows that our approach is better both in the number of subjects and the accuracy of identification. Structural vibration induced by footstep was employed for human identification with a maximum group size of 5 [24]. The accuracy for a group of 5 people was 70%, which is less than our method. After these comparisons, we can see that our results are better or no lower than others.

5. Conclusion

In this paper, we realized human identification based on a heartbeat signal from a radar using a DCNN, SVM, NB and the SVM–Bayes fusion algorithm. For the DCNN, accuracy could reach over 80% when the number of the people was below 10, indicating that humans can be identified successfully by the DCNN, while SVM, NB and SVM–Bayes could reach 88.75%, 80.75%, and 91.25%, respectively, for four people. Additionally, the DCNN had excellent performance when the noise and the number of people increased. By contrast, the performances of the conventional supervised methods were far from satisfactory in both aspects. We believe that it shows the potential of DCNNs for human identification.
identification problems based on the heartbeat signal from radar micro-Doppler signatures. Compared with conventional supervised methods, the amount of data that a DCNN needs is relatively large. In the future, further research will be done for the purpose of improving identification performance. First, the condition we considered in the experiments, where humans sat just 1.5 m in front of the radar, is simple compared with realistic conditions. In the future, we will do further research on data acquisition at long range or non-line of sight (NLOS) scenarios. Second, our experiments only comprised ten people. To apply our system to the real world, we need to include more people. In addition, the optimization for the frame of the network also lies in our future research plan.

**Author Contributions:** Conceptualization, P.C. and W.X.; methodology, validation, formal analysis, data curation, original draft preparation and writing, P.C.; review, editing, and funding acquisition, W.X.; data curation, Y.L.

**Funding:** This research was funded by the National Natural Science Foundation of China (Grant no. 6150010825), the Fundamental Research Funds for the Central Universities (Grant no. NS2016040), the Fundamental Research Funds for the Central Universities (Grant no. NJ20150020), and the China Scholarship Council (Grant no. 201606835062).

**Acknowledgments:** The work was supported by the National Natural Science Foundation of China (Grant no. 6150010825), the Fundamental Research Funds for the Central Universities (Grant no. NS2016040), the Fundamental Research Funds for the Central Universities (Grant no. NJ20150020), and the China Scholarship Council (Grant no. 201606835062).

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

**References**


© 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/).