

# Lightweight Neural Network for Sleep Posture Classification Using Pressure Sensing Mat at Various Sensor Densities

Shaonan Wu, Haikang Diao, Yi Feng, Yiyuan Zhang, Hongyu Chen,  
Yasemin M. Akay, Senior Member, IEEE, Metin Akay, Fellow, IEEE,  
Chen Chen and Wei Chen, Senior Member, IEEE

**Abstract**— Recently, pressure-sensing mats have been widely used to capture static and dynamic pressure over sleep for posture recognition. Both a full-size mat with a low-density sensing array for figuring out the structure of the whole body and a miniature scale mat with a high-density sensing array for identifying the local characteristics around the chest have been investigated. However, both of the mat systems may face the challenge in the trade-off between the computational complexity (involving the size, density, etc. of the mat) and the performance of sleep posture recognition, where high performance may require overcomplex computation and result in time latency in real-time sleep posture monitoring. In this paper, a lightweight neural network named ConcatNet, is proposed to realize sleep postures (supine, left, right, and prone) recognition in real time while maintaining a favorable performance. In ConcatNet, the inception module is proposed to extract the image features under multiple receptive fields, while the multi-layer feature fusion module is utilized to fuse deep and shallow features to enhance the model performance. To further improve the efficiency of the model, the depthwise convolution is adopted. ConcatNet models in 3 different scales (ConcatNet-S, ConcatNet-M, and ConcatNet-L) are built to explore the impact of the sensor density on sleep posture recognition performance. Experimental results exhibit that ConcatNet-M corresponding to medium sensor density ( $16 \times 16$ ) achieved the best comprehensive performance, with short-term data cross-validation accuracy at 95.56% and overnight data testing accuracy at 94.68%. The model size is 7.91KB, FLOPs is 56.47K, and the inference time is only 0.38ms, which shows an outstanding performance of real-time sleep posture recognition with minimum consumption, indicating the potential to be deployed in mobile devices.

**Index Terms**—Sleep posture recognition, real-time, lightweight model, sensor density.

This work was supported by the National Key Research and Development Program of China under Grant 2023YFC3603100 and 2023YFC3603102 (Corresponding authors: Chen Chen and Wei Chen).

Shaonan Wu, Yi Feng, and Yiyuan Zhang are with the Center for Intelligent Medical Electronics, School of Information Science and Technology, Fudan University, Shanghai 200433, China (e-mail: [2121072024@m.fudan.edu.cn](mailto:2121072024@m.fudan.edu.cn); [22210720125@m.fudan.edu.cn](mailto:22210720125@m.fudan.edu.cn); [yamsunshine1992@outlook.com](mailto:yamsunshine1992@outlook.com)).

Haikang Diao is with Institute for Artificial Intelligence and School of Integrated Circuits, Peking University, Beijing 100871, China (e-mail: [diaoхаikang@stu.pku.edu.cn](mailto:diaoхаikang@stu.pku.edu.cn)).

## I. Introduction

SLEEPING is an indispensable part of daily life for human, as it plays crucial roles in brain functions including neurobehavioral, cognitive and safety-related performance [1], etc. In sleep monitoring, sleep posture is an important indicator. To illustrate, staying in the wrong sleep postures can lead to tension in the cervical and lumbar spine, causing pain and even related diseases [2]. In addition, improper sleep postures can cause disturbance in respiration and even sleep apnea, which can lead to death by suffocation in severe cases. What's more, for those patients who are bedridden for long-term have a higher probability of having other disease, such as decubitus ulcers due to long-term pressure [3]. There are around 700000 patients were affected by decubitus ulcers each year [4] and about 60000 people died because of the pressure ulcer complication worldwide [5]. Therefore, it is of great significance for real-time monitoring of sleep posture, not only to alert people to avoid improper sleeping positions but also to remind them not to maintain the same sleep posture for a long time.

The challenges of sleep posture monitoring mainly come from its application scenario, which is long-term and real-time sleep posture monitoring in home environment. Since it involves long-term, real-time monitoring, sleep and private environment (e.g., bedroom), it has higher requirements for the system in terms of robustness, real-time performance, ease of use, comfort and privacy protection. In recent studies, methods on sleep posture monitoring could be mainly classified into 4 categories, including wearable device-based [6], [7], camera-based [8], [9], radio-frequency-based [10], [11], and pressure sensitive mat-based. For wearable device-based methods, the devices attach directly to the human body, which may cause discomfort and interfere with normal sleep. In addition, motion

Hongyu Chen is with the Greater bay area institute of precision medicine, Guangzhou 511466, China. (email: [chenhongyudesign@outlook.com](mailto:chenhongyudesign@outlook.com)).

Yasemin M. Akay and Metin Akay are with the University of Houston, Houston, TX, 77204, USA (e-mail: [ymakay@uh.edu](mailto:ymakay@uh.edu); [makay@uh.edu](mailto:makay@uh.edu)).

Chen Chen is with the Center for Medical Research and Innovation, Shanghai Pudong Hospital, Fudan University Pudong Medical Center, 2800 Gongwei Road, Pudong, Shanghai 201399, China, and also with the Human Phenome Institute, Fudan University, Shanghai 201203, China (e-mail: [chenchen\\_fd@fudan.edu.cn](mailto:chenchen_fd@fudan.edu.cn)).

Wei Chen is with the School of Biomedical Engineering, University of Sydney, Sydney, NSW 2006, Australia (email: [wei.chenbme@sydney.edu.au](mailto:wei.chenbme@sydney.edu.au)).

artifacts can also affect the accuracy of the system. In camera-based systems, the image quality is greatly affected by ambient lighting, and the use of camera in bedrooms may raise privacy concerns. As for radio-frequency-based approaches, some radar devices are expensive and require installation. In addition, the presence of clutter in indoor settings may produce multipath effects [12], and factors such as distance, location, environment, and identity difference may also affect the robustness of the system [13].

In contrast to the aforementioned approaches, pressure sensitive mat-based systems gain the benefits as follows: 1) In the usage scenario, the pressure sensing mat can be placed under the bed sheet, without directly contact with the human body, which ensures the comfort; 2) The system is relatively stable and less affected by the environment (e.g. the change of light, the cover of quilt, position and movement of surrounding objects); 3) The smart mat leads to fewer concerns on privacy issue; 4) The material cost is relatively low.

The initial idea of the pressure sensing mat, starts with building a full-size bed to collect the pressure distribution of the whole human body, then conduct a visualization operation by converting the data into the format of image. The feature of the image could be extracted and classified into different sleep postures by proposing traditional machine learning methods or deep learning methods. In some studies, pressure sensing mats in large scale are deployed [14]-[19]. For example, Xu *et al.* [15] proposed a mat with 8192 (128×64) pressure sensors in 1.62 m<sup>2</sup> area. In [16], a pressure sensor array consists of 1728 (64×27) sensors within the dimensions of 185×76cm region is implemented. And the smart mat used in [17] has 64×32 for a total of 2048 sensors in a bed-size area. These works proved the feasibility of sleep posture classification based on pressure sensing mat. However, when people lie on the large-scale smart mat, the body only covers some of the sensors, which means part of the sensors are not utilized in practical, which may bring redundant information. Besides, large-scale mat has higher production cost, and in most cases, has more sensors which increase the computational complexity. To address above issues, a miniature-scale smart mat implemented by 1024 (32×32) pressure sensors with the size of 55×55cm is proposed in our previous work [20]-[22]. The mat covers the major regions of the chest, part of the shoulders and part of the hips, which is feasible to classify the sleep postures. Liu *et al.* also implement a pressure sensing mat of 100×60cm with 2048 sensors in [23], [24].

In the existing research, from the perspective of hardware, most studies use large-size pressure sensing mats [14]-[19], which have shortcomings like area redundancy and high cost as mentioned earlier. From the perspective of algorithm, some studies have used traditional machine learning algorithms [14]-[16], [21], [23], which have the disadvantages of requiring manual feature extraction and limited recognition capabilities. Another part of the research uses deep learning algorithms [17]-[20], [22], [24]. In order to achieve higher accuracy in sleep posture recognition, most methods use neural networks with higher complexity, which makes it difficult to achieve real-time sleep posture monitoring. Meanwhile, complex models are

difficult to deploy on devices with small capacity and low computing power, so there are certain limitations in corresponding application scenarios. Although some methods use lightweight models, they still need to be used with data calibration [19] or other algorithm modules [22], which will increase the complexity of the algorithm to a certain extent, having negative impact on the real-time performance. In terms of model architecture, [17][19][24] adopted the most common serial CNN structure. The Inception-V3 [18] contains parallel convolutional modules and can extract multi-scale features. The ResNet18 [20] introduced the residual structure, making it possible to build very deep networks. The Tiny-MobileNetV2 [22] also deployed the residual structure while the model is much shallower than ResNet18. While in this paper, the proposed lightweight CNN, ConcatNet contains both parallel convolution structure and multi-layer feature fusion structure, which is mainly designed for the characteristics of the pressure distribution images, and the motivation for the specific architecture will be elaborated in Section II.C.4).

The information, or the feature contained in the pressure distribution image, is mainly affected by 2 factors. The first one has already been motioned above, is the size of the mat, in other words, the body coverage of the mat, which affects the morphology of the image. The second is the sensor density of the mat, which influences the fineness and details of the image. The higher of the sensor density, the more details the image contains. There's no doubt that adequate information is indispensable to classify sleep postures, however, redundant information may have negative influence on classification. For instance, if the sensor density is too high, on the one hand, computational complexity increases sharply, making it hard to implement real-time sleep posture classification; on the other hand, as body shape and weight distribution vary from person to person, some non-universal features brought by high density may be involved in the algorithm, which may reduce the robustness of the classification model. However, in the existing studies [14]-[24], whether in large-scale or small-scale mat, the sensor density has not been explored. Besides, due to the high density of the mat, most of the previous works implemented sleep posture classification of offline instead of real-time, which did not meet the requirement of real-time performance mentioned previously. In this paper, to improve the real-time performance of our smart mat system, we explored the influence of the sensor density on sleep posture classification by proposing a lightweight neural network called ConcatNet in 3 sizes (ConcatNet-S, ConcatNet-M, ConcatNet-L), which correspond to 3 different sensor densities (low-density, medium-density, and high-density), respectively. And the method implements real-time sleep posture classification in a miniature-scale, low-density pressure sensing mat proposed in our previous work [20]. The novelty of this paper is summarized as follows:

- 1) A lightweight deep learning neural network is proposed for real-time sleep posture recognition, which reduces large amounts of model size, computational complexity and time consumption of model inference compared with the existing classical models. At the same time, it's able to achieve

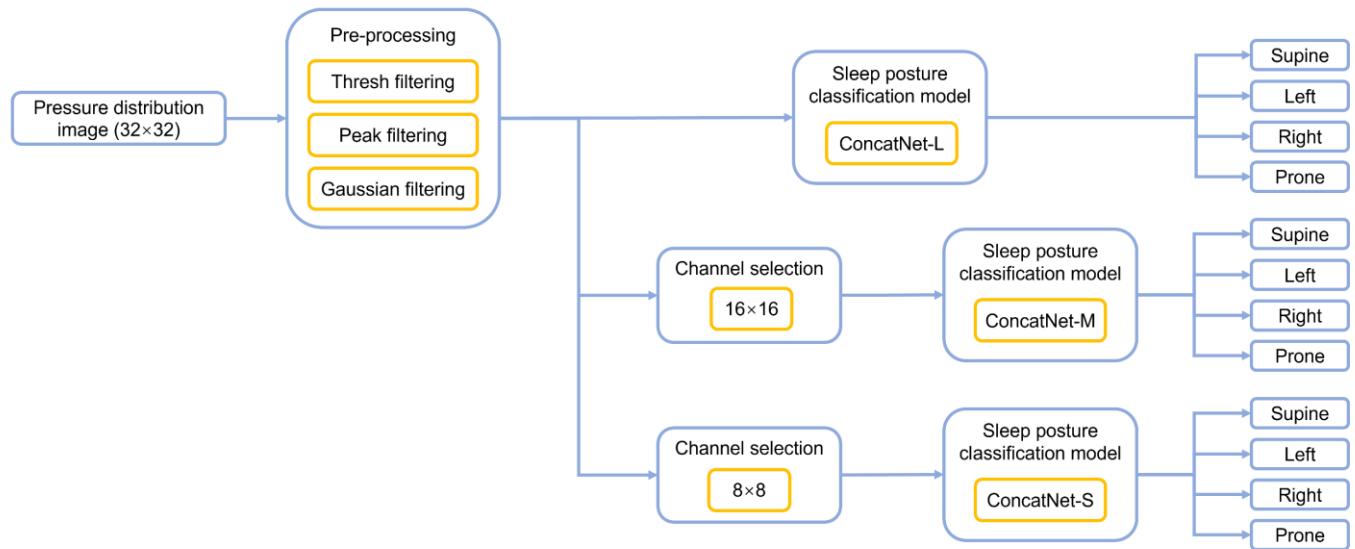


Fig. 1. The algorithm flow of sleep posture classification using pressure distribution image

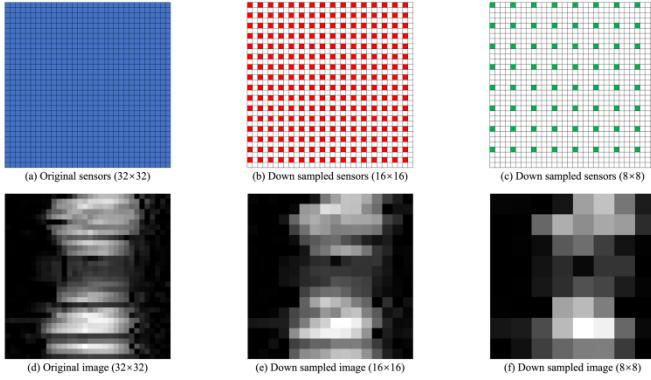


Fig. 2. The process of down sampling. The highlighted pixels in (a), (b), and (c) represent the sampled sensor, and (d), (e), and (f) are the corresponding images of supine posture.

comparable performance, paving the potential way for the deployment on mobile devices.

2) A trade-off between the sensor density of the mat and the performance of sleep posture recognition model is analyzed. The ConcatNet in 3 sizes are implemented, and the comparison between the performance of the neural networks corresponding to different sensor densities are carried out. As a result, it demonstrates the possibility of utilizing a miniature-scale, low-density mat to implement real-time sleep posture classification.

The rest of the paper is organized as follows: Section II presents the algorithm for sleep posture classification, including pre-processing, channel selection and the classification model. Section III shows the results of sleep posture classification with model performance. The discussion and conclusion are presented in Section IV and V, respectively.

## II. ALGORITHM

The algorithm flow is shown in Fig. 1. The raw pressure distribution image in the resolution of 32x32 is first pre-processed to remove the noise, then the image is downsampled

to 16x16 and 8x8 to implement the channel selection. The images in 3 sizes are then sent to the corresponding model to realize sleep posture classification. The advantage of the proposed framework is that the computational complexity is relatively low, as the only steps before sleep posture classification are pre-processing and channel selection, both of which have very little computational cost.

### A. Pre-processing

The raw images of the dataset are firstly processed to reduce the noises, including threshold filtering that removes the background pixels, peak filtering that reduces the influence of adjacent areas, and Gaussian filtering that makes the images smoother. The detailed information on pre-processing can be found in our previous work [20].

### B. Channel selection

In many previous works, due to the large size and high density of the pressure sensing mat, pressure distribution images in high resolution were collected and used directly to train the deep learning model, resulting in a relatively large size of the model and high computational complexity. Higher resolution pressure distribution image contains more details and more information, but for the sleep posture classification task, the extra information contained in the high-resolution images is actually redundant. Therefore, if the size of the pressure distribution images could be reduced, the size and complexity of the network could also be reduced, which can save the time consumption of training and inferencing, making it easier to realize real-time sleep posture monitoring.

In order to implement the channel selection and explore the influence of sensor density on the performance of the sleep posture classification network, the original images with the resolution of 32x32 are down sampled to the resolution of 16x16 and 8x8, by selecting the channels at intervals, which is shown in Fig. 2. Deep neural networks in different sizes are implemented for each corresponding resolution.

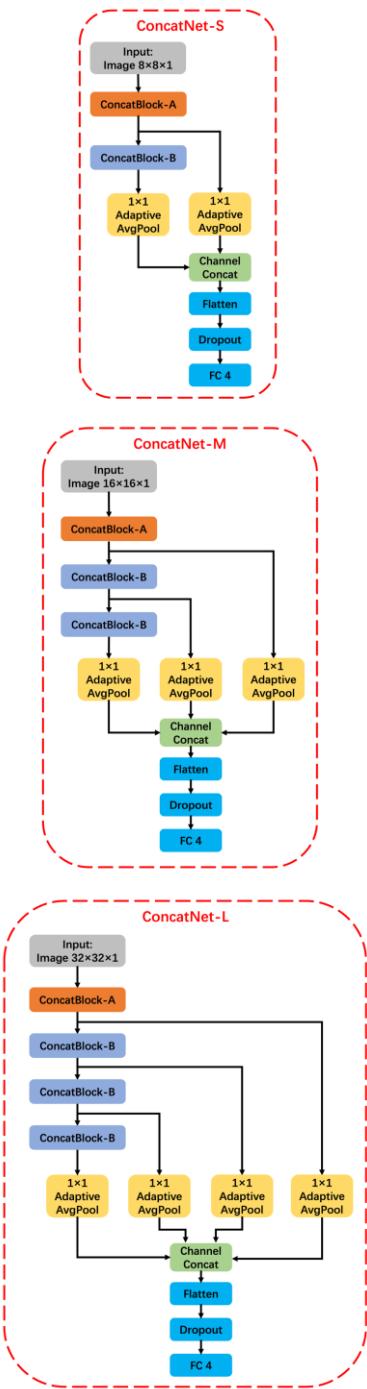


Fig. 3. The structures of ConcatNet in 3 sizes

### C. Sleep posture classification model

To realize real-time sleep posture monitoring, we have designed a lightweight convolutional neural network called ConcatNet. The network gets the name because of the extensive utilization of feature map concatenation operations in the network structure. The overall structure, basic module, optimization measures of the model and the motivation for the architecture will be introduced successively.

#### 1) Model structure

The structure of the proposed model is shown in Fig. 3. The

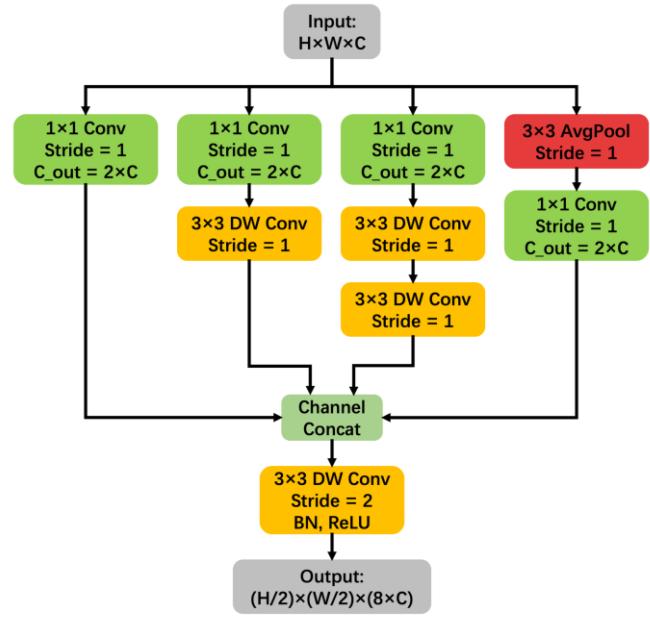


Fig. 4. The structure of ConcatBlock-A

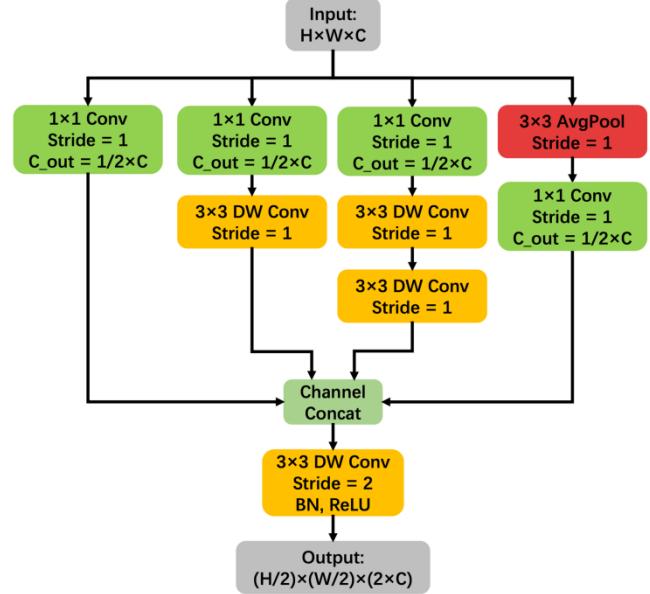


Fig. 5. The structure of ConcatBlock-B

model consists of several basic modules that stacked together, which named as ConcatBlock. The ConcatBlock has two different types, ConcatBlock-A and ConcatBlock-B. Different numbers of Concatblock are stacked in the model for inputs of different sizes. For ConcatNet-L corresponding to image of  $32 \times 32$ , 4 ConcatBlocks (1 ConcatBlock-A, 3 ConcatBlock-B) are stacked, for ConcatNet-M corresponding to image of  $16 \times 16$ , 3 ConcatBlocks (1 ConcatBlock-A, 2 ConcatBlock-B) are stacked, and for ConcatNet-S corresponding to image of  $8 \times 8$ , 2 ConcatBlocks (1 ConcatBlock-A, 1 ConcatBlock-B) are stacked. The output feature maps of each ConcatBlock are globally average pooled and scaled to  $1 \times 1$  in size respectively. After that, these feature maps are concatenated on the channel

TABLE I  
THE ARCHITECTURES OF CONCATNET IN 3 SIZES  
(K=KERNEL SIZE, C=CHANNEL, S=STRIDE, DW=DEPTHWISE CONVOLUTION, CAT=CONCATENATE, GAP=GLOBAL AVERAGE POLLING, AP=AVERAGE POLLING, FC=FULL CONNECTION)

| Layer name    | Input size  |             |             | Layers      | Output size           |             |             |
|---------------|-------------|-------------|-------------|-------------|-----------------------|-------------|-------------|
|               | ConcatNet-L | ConcatNet-M | ConcatNet-S |             | ConcatNet-L           | ConcatNet-M | ConcatNet-S |
| ConcatBlock-A | 32×32×1     | 16×16×1     | 8×8×1       | K=1×1, C=2  | CAT<br>DW, K=3×3, S=2 | 16×16×8     | 8×8×8       |
|               |             |             |             | K=1×1, C=2  |                       |             |             |
|               |             |             |             | DW, K=3×3   |                       |             |             |
|               |             |             |             | K=1×1, C=2  |                       |             |             |
|               |             |             |             | DW, K=3×3   |                       |             |             |
| ConcatBlock-B | 16×16×8     | 8×8×8       | 4×4×8       | K=1×1, C=2  | CAT<br>DW, K=3×3, S=2 | 8×8×16      | 4×4×16      |
|               |             |             |             | K=1×1, C=4  |                       |             |             |
|               |             |             |             | K=1×1, C=4  |                       |             |             |
|               |             |             |             | DW, K=3×3   |                       |             |             |
|               |             |             |             | K=1×1, C=4  |                       |             |             |
| ConcatBlock-B | 8×8×16      | 4×4×16      | -           | DW, K=3×3   | CAT<br>DW, K=3×3, S=2 | 4×4×32      | 2×2×32      |
|               |             |             |             | K=1×1, C=8  |                       |             |             |
|               |             |             |             | K=1×1, C=8  |                       |             |             |
|               |             |             |             | DW, K=3×3   |                       |             |             |
|               |             |             |             | DW, K=3×3   |                       |             |             |
| ConcatBlock-B | 4×4×32      | -           | -           | AP, K=3×3   | CAT<br>DW, K=3×3, S=2 | 2×2×64      | -           |
|               |             |             |             | K=1×1, C=8  |                       |             |             |
|               |             |             |             | K=1×1, C=16 |                       |             |             |
|               |             |             |             | K=1×1, C=16 |                       |             |             |
|               |             |             |             | DW, K=3×3   |                       |             |             |
| GAP, CAT      | 16×16×8     | -           | -           | K=1×1, C=16 | CAT                   | 1×1×120     | 1×1×56      |
|               | 8×8×16      | 8×8×8       | -           | AP, K=16×16 |                       |             |             |
|               | 4×4×32      | 4×4×16      | 4×4×8       | AP, K=8×8   |                       |             |             |
|               | 2×2×64      | 2×2×32      | 2×2×16      | AP, K=4×4   |                       |             |             |
| Flatten, FC   | 1×1×120     | 1×1×56      | 1×1×24      | AP, K=2×2   | FC, 4                 | 1×4         |             |

dimension, and then flattened and sent into the full connection layer for classifying into 4 categories of sleep postures.

### 2) Basic module – ConcatBlock

As mentioned earlier, the basic module of the model is the ConcatBlock in 2 types, ConcatBlock-A and ConcatBlock-B. The structure of ConcatBlock-A is shown in Fig. 4.

ConcatBlock-A mainly consists of two parts. The first part is the Inception module originated from the Inception-v3 network [25]. Assuming that the size of the input feature map is  $H \times W \times C$  ( $H$  for height,  $W$  for width, and  $C$  for channel). First, a parallel convolution of 4 branches is performed. The first branch uses a convolution with the kernel size of  $1 \times 1$ , step length of 1, and kernel number of  $2 \times C$ . Based on the first branch, the second branch adds a depthwise convolution whose kernel size is  $3 \times 3$  and step length is 1. The depthwise convolution proposed here is to reduce the model size and computational complexity, which will be discussed in the next part. The third branch adds another depthwise convolution with the kernel size of  $3 \times 3$  and step length of 1 based on the second branch. And the fourth branch uses an average pooling layer with the kernel size of  $3 \times 3$ , step length of 1, and a convolution layer with the kernel size of  $1 \times 1$ , step length of 1, and kernel number of  $2 \times C$ . The size of the output feature map of each branch is  $H \times W \times (2 \times C)$ , and then the output feature maps of the four branches are concatenated in the channel dimension, so the size of the obtained feature map is

$H \times W \times (8 \times C)$ . The feature map is then sent into the second part of ConcatBlock, which is the feature down sampling module composed of depthwise convolution, Batch Normalization (BN) and Rectified Linear Unit (ReLU) activation function. The size of the convolution kernel is  $3 \times 3$  and the step length is 2, so the final output feature map size of the module is  $(H/2) \times (W/2) \times (8 \times C)$ .

The structure of ConcatBlock-B is similar to ConcatBlock-A, as shown in Fig. 5. The only difference is that the kernel number of the convolutions in ConcatBlock-B is reduced to  $(1/2) \times C$ , so the size of the output feature map is  $(H/2) \times (W/2) \times (2 \times C)$ .

As we can see in Fig. 3, in the 3 networks, ConcatBlock-A is set as the first block. The input pressure distribution image is a single channel image, we use ConcatBlock-A to extract the features and expand the channel dimension to 8. For the rest of the ConcatBlocks, we set them as ConcatBlock-B, whose expand ratio of the depth of the feature map is 2 instead of 8 for ConcatBlock-A. By this way, the expansion speed of channel dimension is slower and the computation cost could be reduced.

The architectures of 3 ConcatNets are shown in Table I.

### 3) Lightweight method – Depthwise convolution

As mentioned above, in the ConcatBlock, depthwise convolution is proposed in the second and third branch of the parallel convolution, and the down sampling module. As the

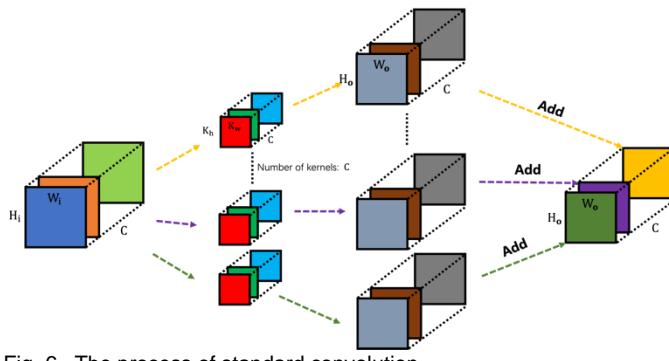


Fig. 6. The process of standard convolution

name implies, depthwise convolution is to apply a single channel convolution kernel to each channel of the input feature map for a convolution operation in the depth direction, that is, in the channel dimension. Then, the obtained feature map is concatenated together in the channel dimension to obtain the output feature map. Thus, the number of channels of the input feature map and the output feature map are consistent. Depthwise convolution has been applied in classic network models such as MobileNet [26] and Xception [27]. Compared with standard convolution, depthwise convolution can reduce the number of parameters and the computational complexity. Fig. 6 and Fig. 7 show the process of standard convolution and depthwise convolution respectively.

Suppose the size of the input feature map is  $H_i \times W_i \times C$ , the standard convolution includes a convolution kernel with kernel size of  $K_h \times K_w \times C$  and kernel number of  $C$ , while the convolution kernel in depthwise convolution has a kernel size of  $K_h \times K_w \times 1$  and kernel number of  $C$ . The size of the output feature map is  $H_o \times W_o \times C$ , which is the same for standard convolution and depthwise convolution. Ignore the bias of the convolution, the number of parameters for standard convolution is:

$$K_h \cdot K_w \cdot C \cdot C \quad (1)$$

The number of parameters for depthwise convolution is:

$$K_h \cdot K_w \cdot C \quad (2)$$

The ratio of the number of parameters of depthwise convolution to standard convolution is:

$$\frac{K_h \cdot K_w \cdot C}{K_h \cdot K_w \cdot C \cdot C} = \frac{1}{C} \quad (3)$$

The multiplication and addition operations of standard convolution is:

$$H_o \cdot W_o \cdot C \cdot (2 \cdot C \cdot K_h \cdot K_w - 1) \quad (4)$$

The multiplication and addition operations of depthwise convolution is:

$$H_o \cdot W_o \cdot C \cdot (2 \cdot K_h \cdot K_w - 1) \quad (5)$$

The ratio of multiplication and addition operations between depthwise convolution and standard convolution is:

$$\frac{H_o \cdot W_o \cdot C \cdot (2 \cdot K_h \cdot K_w - 1)}{H_o \cdot W_o \cdot C \cdot (2 \cdot C \cdot K_h \cdot K_w - 1)} = \frac{1}{C - 1} \leq \frac{1}{C} \quad (6)$$

As Table I shows,  $C = 2, 4, 8, 16$  in different stages. Thus, compared with standard convolution, depthwise convolution significantly reduces the number of parameters and the computational complexity.

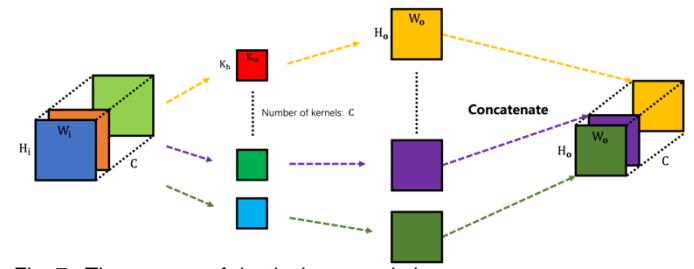


Fig. 7. The process of depthwise convolution

#### 4) Motivation for the architecture of ConcatNet

TABLE II  
INFORMATION OF SUBJECTS IN SHORT-TERM EXPERIMENT

| Subject | Gender | Age | Height (cm) | Weight (kg) | Number of samples |
|---------|--------|-----|-------------|-------------|-------------------|
| 1       | Female | 23  | 158         | 55          | 59                |
| 2       | Male   | 31  | 178         | 76.5        | 72                |
| 3       | Male   | 22  | 171         | 76          | 72                |
| 4       | Male   | 26  | 170         | 65          | 72                |
| 5       | Female | 24  | 168         | 50          | 68                |
| 6       | Male   | 25  | 175         | 60          | 72                |
| 7       | Female | 23  | 158         | 48          | 72                |
| 8       | Male   | 24  | 171         | 70          | 71                |
| 9       | Female | 28  | 165         | 52          | 72                |
| 10      | Female | 29  | 160         | 51          | 72                |
| 11      | Male   | 25  | 173         | 75          | 72                |
| 12      | Male   | 29  | 175         | 83          | 60                |
| 13      | Male   | 23  | 180         | 70          | 72                |
| 14      | Female | 26  | 170         | 54          | 69                |
| 15      | Female | 28  | 160         | 50          | 45                |
| 16      | Male   | 33  | 174         | 75          | 39                |

In the previous paragraphs, the architecture of ConcatNet is introduced in detail. The motivation for the design of the architecture mainly comes from the characteristics of the pressure distribution images. Very different from natural images, the characteristics of pressure distribution images include: low resolution, relatively fixed image content, and small data quantity. In view of these characteristics, the design of the model needs to take into account the efficiency of feature extraction and utilization as well as the complexity and real-time performance of the model.

In terms of feature extraction, considering the image characteristics, it is not suitable to use an overly deep network structure because overfitting is prone to occur. However, using a shallow network may also lead to insufficiency of the extracted features. Therefore, ConcatNet uses the Inception module to improve the model's ability to extract multi-scale features without increasing the depth of the network. Considering feature utilization, we refer to the idea of fusing deep and shallow features in the U-Net [28] model and add a deep and shallow feature fusion module to ConcatNet. The reason is that U-Net has a very good performance in extracting the features of medical images, which are similar to pressure distribution images in characteristics mentioned above. Finally, take the complexity and real-time performance of the model

TABLE III  
POSTURE CLASSIFICATION ACCURACY OF EACH SUBJECT

| Subject | ConcatNet-S  | ConcatNet-M  | ConcatNet-L  |
|---------|--------------|--------------|--------------|
| 1       | 78.0%        | 93.2%        | 98.3%        |
| 2       | 87.5%        | 94.4%        | 100.0%       |
| 3       | 84.7%        | 94.4%        | 84.7%        |
| 4       | 90.3%        | 90.3%        | 100.0%       |
| 5       | 88.2%        | 83.8%        | 91.2%        |
| 6       | 98.6%        | 100.0%       | 100.0%       |
| 7       | 91.7%        | 95.8%        | 100.0%       |
| 8       | 94.4%        | 95.8%        | 100.0%       |
| 9       | 86.1%        | 100.0%       | 100.0%       |
| 10      | 90.3%        | 100.0%       | 100.0%       |
| 11      | 100.0%       | 100.0%       | 100.0%       |
| 12      | 81.7%        | 91.7%        | 96.7%        |
| 13      | 90.3%        | 100.0%       | 100.0%       |
| 14      | 97.1%        | 100.0%       | 100.0%       |
| 15      | 91.1%        | 97.8%        | 91.1%        |
| 16      | 71.8%        | 87.2%        | 97.4%        |
| Average | <b>89.5%</b> | <b>95.6%</b> | <b>97.6%</b> |

into consideration. On the one hand, the aforementioned avoidance of using too deep network has reduced the complexity of the model to a certain extent. On the other hand, we use deep convolution in ConcatNet to further reduce the complexity of the model, which also improves real-time performance.

### III. RESULTS

A total of 16 subjects (9 males, 7 females) were included in the short-term experiment and 5 male subjects were included in the overnight experiment, as described in our previous work [20]. The first part (Dataset A) is the short-term experiment data that consists of 1059 pressure distribution images and corresponding sleep posture labels. During the short-term experiment, all subjects were instructed to lie on the mat with 4 different sleep postures: supine, right, left, and prone. Each posture was held for 20 seconds, and the interval between the two postures was 15 seconds. The sampling rate is 0.1 Hz, and about 2 images are saved within 20 seconds. In order to simulate the real scenario, the subjects can move slightly during the experiment. Some subjects may move significantly during the experiment, then these samples will be discarded. Therefore, the number of samples for each subject is different. The information of each subject is given in Table II. The second part (Dataset B) comes from the overnight experiment conducted in the laboratory with whole nights of continuous sleep. 5 male subjects participated in the overnight experiment for 7 nights. An infrared camera continuously recorded the ground truth postures. During the experiments, all the sleeping stages were covered. After excluding the time of turning over, 57 hours of sleeping data was recorded and 20521 pressure distribution images were collected in total. As the size of the sensor array is 32×32, the resolution of the pressure distribution images is also

TABLE IV  
POSTURE CLASSIFICATION RESULT OF CONCATNET-S

| Predicted | Ground Truth  |        |        |        |           |
|-----------|---------------|--------|--------|--------|-----------|
|           | Supine        | Right  | Left   | Prone  | Precision |
| Supine    | 223           | 3      | 4      | 22     | 89.68%    |
| Right     | 8             | 259    | 13     | 1      | 97.28%    |
| Left      | 12            | 4      | 236    | 7      | 94.45%    |
| Prone     | 29            | 1      | 7      | 230    | 89.06%    |
| Recall    | 81.99%        | 97.00% | 90.77% | 88.80% |           |
| Accuracy  | <b>89.52%</b> |        |        |        |           |

| TABLE V<br>POSTURE CLASSIFICATION RESULT OF CONCATNET-M |               |        |        |        |           |
|---|---------------|--------|--------|--------|-----------|
| Predicted   | Ground Truth  |        |        |        |           |
|   | Supine        | Right  | Left   | Prone  | Precision |
| Supine  | 251           | 4      | 5      | 6      | 94.36%    |
| Right   | 4             | 260    | 0      | 5      | 96.65%    |
| Left  | 7             | 3      | 254    | 2      | 95.49%    |
| Prone   | 10            | 0      | 1      | 247    | 95.74%    |
| Recall  | 92.28%        | 97.38% | 97.69% | 95.00% |           |
| Accuracy  | <b>95.56%</b> |        |        |        |           |

TABLE VI  
POSTURE CLASSIFICATION RESULT OF CONCATNET-L

| Predicted | Ground Truth  |        |        |        |           |
|-----------|---------------|--------|--------|--------|-----------|
|           | Supine        | Right  | Left   | Prone  | Precision |
| Supine    | 267           | 2      | 3      | 3      | 97.09%    |
| Right     | 0             | 260    | 0      | 2      | 99.24%    |
| Left      | 0             | 3      | 254    | 2      | 98.07%    |
| Prone     | 5             | 2      | 3      | 253    | 96.20%    |
| Recall    | 98.16%        | 97.38% | 97.69% | 97.31% |           |
| Accuracy  | <b>97.64%</b> |        |        |        |           |

32×32.

#### A. Cross-validation on short-term dataset

As mentioned above, in the short-term experiment, subjects were instructed to lie on the pressure sensing mat in different sleep postures for data collection. The main purpose of collecting this part of data is to train and verify the sleep posture classification model. Since the short-term dataset contains only 1059 images of pressure distribution images, which is in a relatively small scale. In order to maximize the utilization of the data, Leave-One-Subject-Out cross-validation (LOSOCV) is adopted. The dataset contains 16 subjects, so the number of folds equals 16, in other words, the dataset is divided into 16 parts according to each subject, and a total of 16 rounds of training are performed. During each training round, all the data of one subject is selected as the verification set, and the data of the remaining 15 subjects is collectively taken as the training set. The data of each subject is used as a training set and a validation set in different training rounds, so the data is utilized to the maximum. After completing each round of training, the classification results of the model on the verification set are

TABLE VII  
METRICS OF DIFFERENT MODELS ON THE SHORT-TERM AND OVERNIGHT DATASET

| Input size                           | 8×8           | 16×16         | 32×32          |            |              |             | 160×160  |          |
|--------------------------------------|---------------|---------------|----------------|------------|--------------|-------------|----------|----------|
| Model                                | ConcatNet-S   | ConcatNet-M   | ConcatNet-L    | MobileNeXt | ShuffleNetV2 | MobileNetV2 | ResNet18 | Xception |
| Short-term cross-validation accuracy | <b>89.52%</b> | <b>95.56%</b> | <b>97.64%</b>  | 95.94%     | 90.56%       | 88.10%      | 96.98%   | 94.81%   |
| Overnight testing accuracy           | <b>88.82%</b> | <b>94.68%</b> | <b>92.91%</b>  | 83.23%     | 82.95%       | 85.41%      | 85.00%   | 88.28%   |
| Number of parameters                 | <b>0.73K</b>  | <b>2.02K</b>  | <b>5.64K</b>   | 0.12M      | 0.35M        | 2.31M       | 11.17M   | 20.94M   |
| Model size                           | <b>2.84KB</b> | <b>7.91KB</b> | <b>22.03KB</b> | 0.47MB     | 1.32MB       | 8.79MB      | 42.62MB  | 79.87MB  |
| FLOPs                                | <b>10.74K</b> | <b>56.47K</b> | <b>0.27M</b>   | 1.23M      | 0.78M        | 6.59M       | 890.23M  | 2.33G    |
| Inference time                       | <b>0.36ms</b> | <b>0.38ms</b> | <b>0.39ms</b>  | 0.49ms     | 0.50ms       | 0.49ms      | 2.21ms   | 5.13ms   |

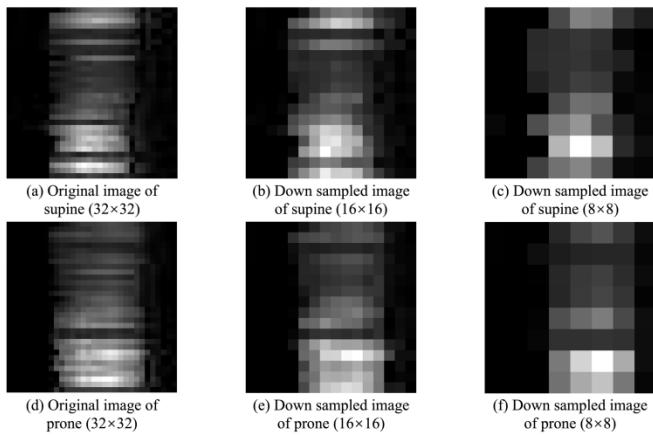


Fig. 8. The difference between pressure distribution images of supine and prone. (a)(b)(c) show a supine image collected by the pressure sensor array in high density, medium density, and low density, respectively, (d)(e)(f) show a prone image collected by the pressure sensor array in high density, medium density, and low density, respectively.

collected, and after all 16 rounds of training, the results are summarized. The performance of the network would be analyzed, and the model would be optimized for the testing according to the verification results. Finally, the relevant hyperparameters are set as follows: the number of epochs of training is 100, the batch size is 64, the optimizer is adaptive moment estimation (Adam), the learning rate is 0.001, the loss function is categorical cross-entropy. The platform is a Windows 10 PC using PyTorch 1.13 with CPU of INTEL i5-8400 and GPU of NVIDIA GTX1060.

Table III shows the sleep posture classification accuracy of 3 ConcatNet models in different sizes on each subject. For the data of 16 subjects, the models corresponding to different sensor densities all perform well on classifying the sleep postures, and the classification accuracy of the small-scale model (ConcatNet-S) ranges from 71.8% to 100%, with an average accuracy of 89.5%. The accuracy of medium-scale model (ConcatNet-M) ranges from 83.8% to 100%, and the average accuracy is 95.6%. The classification accuracy of the large-scale model (ConcatNet-L) ranges from 84.7% to 100%, with an average accuracy of 97.6%. It is not difficult to find that, with the increase of sensor density, classification accuracy in cross-validation also increases.

Table IV-VI show the posture classification results of ConcatNet-S, ConcatNet-M, and ConcatNet-L, respectively. In Table IV, as we can see clearly, ConcatNet-S performs better at distinguishing left and right sleep postures than supine and prone postures, which is mainly caused by the subtle difference between the images of supine and prone in low resolution. As shown in Fig. 8, the difference between supine and prone images in 8×8 resolution is less noticeable than the higher resolution ones. With images in higher resolution and details, ConcatNet-M and ConcatNet-L both get balanced performance on all 4 postures and achieve satisfying results, as shown in Table V and VI.

### B. Testing on overnight dataset

In order to verify the performance of our model in real scenario, dataset A is utilized as the training set, and dataset B as the testing set. The hyperparameters selected in the cross-validation process are used to train the model. After the testing, the metrics of the model are calculated.

The metrics consist of two parts, which are, the accuracy metrics and the complexity metrics. The accuracy metrics include the cross-validation accuracy on short-term dataset and the testing accuracy on overnight dataset, which indicate whether the model is accurate in recognizing sleep postures. While the complexity metrics consist of number of parameters, model size, Floating-point Operations (FLOPs), and inference time. Number of parameters refers to the amount of parameters in the model structure. Model size refers to the amount of space required by computers and other devices to store model parameters. Model size is positively related to number of parameters and the ratio between the two depends on the data type of the parameters. Number of parameters and model size can reflect the model's consumption of storage space resources. Excessive number of parameters and model size will limit its application in certain scenarios, such as the deployment of the model on embedded devices with small storage capacity. FLOPs are used to describe the overall calculation amount of the model, including the calculation amount generated by convolutions, poolings, activation functions, etc. in the model structure. Inference time refers to the time it takes for the model to process a single piece of input data. FLOPs and inference time reflect the model's consumption of computing resources. The longer the inference time, the worse the real-time

TABLE VIII  
COMPARISON OF THE PROPOSED METHOD WITH EXISTING METHODS (M: NUMBER OF SUBJECTS; N: NUMBER OF SAMPLES; N/A: NOT MENTIONED)

| Methods                         | Dataset             | Sensor Channels | Area and Sensor Density                      |   | Algorithms                         | Postures      | Accuracy      |  |
|---------------------------------|---------------------|-----------------|--|---|------------------------------------|---------------|---------------|--|
| Tang <i>et al.</i> , 2021 [18]  | M=N/A, N=1200       | 9×19            | Large-scale, low-density                     | 2m <sup>2</sup> , 0.0086/cm <sup>2</sup>      | InceptionV3 with transfer learning | 4             | 90.00%        |  |
| Sun <i>et al.</i> , 2016 [14]   | M=15, N=N/A         | 128×48          | Large-scale, high-density                    | 1.55m <sup>2</sup> , 0.40/cm <sup>2</sup>     | Weighted limb clusters             | 6             | 97.80%        |  |
| Xu <i>et al.</i> , 2016 [15]    | M=14, N=1848        | 128×64          |  | 1.62m <sup>2</sup> , 0.51/cm <sup>2</sup>     | EMD+K nearest neighborhood         | 6             | 91.21%        |  |
| Matar <i>et al.</i> , 2020 [16] | M=12, N=1116        | 64×27           | Large-scale, medium-density                  | 1.41m <sup>2</sup> , 0.12/cm <sup>2</sup>     | Feature extraction + FFANN         | 4             | 97.90%        |  |
| Hu <i>et al.</i> , 2021 [19]    | M=5, N=7000         | 32×32           |  | 1.62m <sup>2</sup> , 0.063/cm <sup>2</sup>    | CNN with transfer learning         | 6             | 91.24%        |  |
| Diao <i>et al.</i> , 2021 [20]  | M=16, N=1059        | 32×32           | Miniature-scale, high-density                | 0.3m <sup>2</sup> , 0.34/cm <sup>2</sup>      | ResNet18                           | 4             | 95.08%        |  |
|                                 | M=7, N=20521        |                 |  | 0.3m <sup>2</sup> , 0.34/cm <sup>2</sup>      |                                    |               | 86.35%        |  |
| Diao <i>et al.</i> , 2021 [21]  | M=7, N=224          | 32×32           |  | 0.3m <sup>2</sup> , 0.34/cm <sup>2</sup>      | KNN with transfer learning         | 4             | 96.43%        |  |
| Diao <i>et al.</i> , 2022 [22]  | M=16, N=1059        | 32×32           |  | 0.3m <sup>2</sup> , 0.34/cm <sup>2</sup>      | FCSNet + Tiny-MobileNetV2          | 4             | 95.43%        |  |
|                                 | M=7, N=20521        |                 |  | 0.3m <sup>2</sup> , 0.34/cm <sup>2</sup>      |                                    |               | 86.80%        |  |
| Proposed                        | <b>M=16, N=1059</b> | 32×32           | <b>0.3m<sup>2</sup>, 0.34/cm<sup>2</sup></b> | <b>ConcatNet-L</b>                            | 4                                  | <b>97.64%</b> | <b>97.64%</b> |  |
|                                 | <b>M=7, N=20521</b> |                 |  |   |                                    |               | <b>92.91%</b> |  |
| Proposed                        | <b>M=16, N=1059</b> | 16×16           | Miniature-scale, medium-density              | <b>0.3m<sup>2</sup>, 0.085/cm<sup>2</sup></b> | ConcatNet-M                        | 4             | <b>95.56%</b> |  |
|                                 | <b>M=7, N=20521</b> |                 |  |   |                                    |               | <b>94.68%</b> |  |
| Proposed                        | <b>M=16, N=1059</b> | 8×8             | Miniature-scale, low-density                 | <b>0.3m<sup>2</sup>, 0.021/cm<sup>2</sup></b> | ConcatNet-S                        | 4             | <b>89.52%</b> |  |
|                                 | <b>M=7, N=20521</b> |                 |  |   |                                    |               | <b>88.82%</b> |  |

performance of the model. For devices with poor computing performance, excessive FLOPs will seriously affect the speed of the model and have restrictions on scenarios with real-time requirements.

The results are shown in Table VII. The size of the original images in the dataset is 32×32, and for some large models, we up sample the original images to the resolution of 160×160 to match the model structures as [20] proposed.

Benefiting from the simple design of the network structure and the utilization of depthwise convolution, the model size, computational complexity and inference time have been significantly reduced. The ConcatNet in 3 sizes (ConcatNet-S, ConcatNet-M, and ConcatNet-L) have achieved similar or even higher accuracy in sleep posture classification than other classical models in Table VII, both in short-term dataset cross-validation and in overnight dataset testing. With fewer parameters and less computational complexity, faster inference speed and higher classification accuracy are obtained, which shows the great advantages of our proposed model. Considering the reason, we believe that, for pressure distribution image, the shallow feature and deep feature are both crucial for distinguishing different sleep postures. The shallow feature mainly contains local information, such as edge, outline, texture, etc. which could be used to make a rough classification. And the deep feature contains the global semantic information. In ConcatNet, we have fused the shallow feature and the deep feature, while other classic models mainly use the deep feature, that would be the reason why ConcatNet can achieve higher accuracy with less computational cost. Among 3 sizes of ConcatNet, the model size, computational complexity and inference time are all increased with the increase of input size, or sensor density, which is in the expectation. In terms of accuracy, the cross-validation accuracy of short-term data is

also improved with the increase of input size. But regarding the accuracy of the overnight dataset, ConcatNet-M > ConcatNet-L > ConcatNet-S. This indicates that the ConcatNet-L model may have got some overfitting. In other words, the information contained in the input image with 32×32 resolution (high-density) is redundant, and the noise introduced by the higher resolution is also learned by the model. As a result, the cross-validation accuracy of ConcatNet-L is higher than ConcatNet-M while the test set accuracy is lower. Considering both the computational consumption and the sleep posture classification accuracy, the sensor array in medium density with ConcatNet-M has the best comprehensive performance.

The algorithm proposed in this article can be applied in the following aspects. First, the channel selection method in Section II.B can be applied to high-density array sensors. By adjusting the instructions of the acquisition system, interval sampling of sensors can be easily realized, which can reduce the sampling time of pressure distribution data and the complexity of data processing, while still achieving favorable results in tasks such as sleep posture classification. Second, as ConcatNet consumes very little storage space resources (number of parameters, model size) and computing resources (computational complexity), and has strong real-time performance, the model can be deployed to mobile devices or embedded devices instead of being limited to expensive high-performance devices. On the one hand, it can reduce the cost of the monitoring system and facilitate its promotion to home monitoring. On the other hand, it can realize local processing of data and achieve real-time monitoring.

#### IV. DISCUSSION

In this paper, a neural network called ConcatNet, with the characteristics of fast, efficient, and lightweight, is designed and applied to classify 4 sleep postures. The results prove that the model is capable to classify sleep postures in real time with different sensor densities of pressure sensing mat.

##### A. Comparison with the existing works

In comparison with the existing works, Table VIII presents the studies of sleep posture classification based on pressure sensing mat. In terms of hardware, as mentioned earlier, a smaller scaled mat was deployed compared with prior works, and we further explored the impact of the sensor density on sleep posture classification by down sampling the sensor channels to  $16 \times 16$  and  $8 \times 8$ , which correspond to medium-density and low-density mat in a miniature scale, respectively.

In the design of algorithm, in some studies, such as [14]-[16], [21], traditional machine learning methods have been adopted, while [18]-[20], [22] have proposed deep learning methods. Compared with conventional machine learning algorithms, Convolutional Neural Networks (CNN) can eliminate manual feature selection and extraction while achieving better performance in computer vision domain. In studies using CNN, the Inception-v3 in [18] and ResNet18 in [20] are relatively large networks with a slow inference speed, which cannot implement sleep posture classification in real time. Hu *et al.* [19] deployed a simple CNN and achieved real-time classification. However, the accuracy is only 91.24% with transfer learning, which means, part of one subject's data was used for training, while the other part was used for testing, and this also means additional training is required before use in real world scenario. The Tiny-MobileNetV2 in [22] is a lightweight model, however, before the model inferencing, the pressure distribution images need to be processed by the FCSNet, which is a complex frequency channel selection operation with negative influence on real-time performance. In this paper, we proposed a lightweight CNN called ConcatNet. As can be seen from Table VII, compared with other models, ConcatNet has smaller model size, less computational operations and faster inference speed, while achieving high accuracy, which proves that the model can implement real-time sleep posture classification at a high accuracy rate.

##### B. Influence of sensor density on sleep posture classification

Different from previous studies, in this paper, we explored the effect of sensor density on the performance of sleep posture classification by obtaining datasets of different sensor densities through down sampling channels of sensor array and designing a lightweight neural network.

Obviously, with a fixed size, the higher of the sensor density, the higher the resolution of the collected pressure distribution image and the greater the amount of information in it. Therefore, more complex models are needed for machine learning, and theoretically, the higher the accuracy of sleep posture classification should be. However, the results in Table VII are not completely consistent with what were expected. Among the ConcatNet in 3 sizes, with the increase of sensor density, for the model, the number of parameters, model size, computational complexity, and inference time all increase. However, as regard to the classification accuracy, although the accuracy of cross-

validation of short-term data also gradually increases, the accuracy of test set increases first and then decreases. The most reasonable explanation is that in machine learning, we generally assume that the training set and the test set come from the same distribution, that is, the features of the training set data and the test set data are similar, so, if the features of the training set are learned and used to distinguish different categories of data in the test set, which will usually get favorable results. However, in practical application scenarios, the distributions of training set and test set are not exactly consistent. For the dataset used in this paper, factors such as different body shape and weight distribution of individuals in the training set and the test set may lead to differences in data distribution. As mentioned earlier, the higher the resolution of an image, the more information and detail it contains, this additional information is often where the difference in features lies. The model learns too many features of the training set that do not apply to the test set, resulting in overfitting, leading to a lower test set accuracy for ConcatNet-L than for ConcatNet-M. Of course, the above situation holds true when the image resolution has reached the stage to be able to extract enough features for classification. For example, the accuracy of ConcatNet-M is obviously better than that of ConcatNet-S, precisely because the information contained in the image with  $8 \times 8$  resolution is not enough to extract enough features to achieve a favorable classification accuracy.

Regarding the overfitting problem of ConcatNet-L mentioned above, we have tried some methods to alleviate the issue, such as, using data augmentation, adding dropout in the full connection layer of the model, adding L2 regularization in the loss function. The following configuration is adopted, 1) data augmentation is performed on the training set, including random rotation from  $-20^\circ$  to  $20^\circ$  and perspective transformation with a distortion scale of 0.3 and a probability of 0.5, 2) dropout with a probability of 0.3 is added to the full connection layer of the model, 3) L2 regularization with a weight of 0.3 is added to the loss function. With these strategies, the short-term cross-validation accuracy of ConcatNet-L dropped from 97.56% to 96.88%, and the overnight test accuracy increased from 92.57% to 94.59%, which shows that overfitting problem has been mitigated.

Based on the above analysis, we can conclude that in this dataset, the ConcatNet model proposed in this paper can achieve terrific sleep posture classification result in real time with minimal resource consumption. In addition, for the pressure sensing mat system implemented in [20], the data collected by the  $16 \times 16$  channel sensor array (medium density) is sufficient to classify sleep postures. Higher sensor density will not only increase the computational complexity, but also may have a negative impact on the classification accuracy.

In terms of applicable range, this study still has some limitations. For example, although a miniature scaled mat can achieve efficient and accurate sleep posture monitoring for a single person, it cannot be directly applied to scenario of double person, which requires a wider mat. In addition, the pressure distribution images of  $8 \times 8$  channels are difficult to achieve a high classification accuracy, especially in distinguishing between supine and prone.

Some possible research directions include the following points. First, only 3 different sensor densities are tested in this paper. In the future, the types of resolution could be more

refined to further explore its impact on model performance. Second, the algorithm in this paper currently only considers the recognition of sleep posture based on pressure sensing mat. In the future, signals such as body movement and respiratory can also be added to study lightweight algorithms for sleep monitoring based on multi-modal signals. The influence of sensor density on multi-modal signals and optimal sensor selection for adaptation based on task type could also be explored. Third, although the proposed ConcatNet has achieved high generalizability in the sleep posture classification task of different subjects, for the fixed users, combining the model with transfer learning method can further improve the classification accuracy, which is also a research direction of the future.

## V. CONCLUSION

In this paper, a new lightweight neural network model, named as ConcatNet, is proposed, which adopts modularized design and the model series are constructed in different sizes corresponding to different sensor densities. By using the pressure distribution image collected by the pressure sensor array as input, four categories of sleep posture are classified. Comparing the model size, number of parameters, computational complexity, and inference speed with classical models such as ResNet [29] and MobileNet, ConcatNet achieves the best performance. The short-term data cross-validation accuracy of ConcatNet-L with  $32 \times 32$  input size reached 97.64%. The ConcatNet-M corresponding to input size of  $16 \times 16$  achieved the accuracy of 94.68% in overnight data testing. In addition, this paper also explores the influence of the sensor density on the model performance, the results show that ConcatNet-M for medium-density mat has the best comprehensive performance, with the accuracy of short-term data at 95.56%, and the overnight data at 94.68%. The model size is 7.91KB, FLOPs is 56.47K, and the inference time is only 0.38ms, which have realized sleep posture classification in real time. We believe that the system consisting of a miniature-scale, low-density pressure sensing mat and ConcatNet, has great potential to be implemented on mobile devices to realize real-time sleep posture classification. In real-life scenarios, for example, in the monitoring scenario for long-term bedridden patients, real-time and long-term monitoring of the patient's sleep posture through the algorithm proposed in this paper can enable caregivers to know the sleep posture status of the patients and whether they need to help the patients turn over to reduce the risk of occurrence of diseases such as decubitus ulcers. In daily home health monitoring scenarios, by deploying the proposed algorithm to mobile devices or embedded devices, local real-time sleep posture monitoring can be achieved, which can help users understand their own sleep status. Moreover, the system could be improved by integrating other sensors, or connecting with other sleep monitoring devices to obtain more diverse information. Correspondingly, the algorithms could be enhanced to process the multi-modal information and provide more functions. For example, by combining the information of sleep quality analysis, recommendations about which sleep posture will be good to improve sleep quality on a personalized basis could be given in the future.

## REFERENCES

- [1] N. F. Watson, M. S. Badr, G. Belenky, D. L. Bliwise, O. M. Buxton, D. Buysse, D. F. Dinges, J. Gangwisch, M. A. Grandner, C. Kushida, R. K. Malhotra, J. L. Martin, S. R. Patel, S. F. Quan, E. Tasali, M. Twery, J. B. Croft, E. Maher, J. A. Barrett, S. M. Thomas, and J. L. Heald, "Recommended amount of sleep for a healthy adult: A joint consensus statement of the american academy of sleep medicine and sleep research society," *Journal of Clinical Sleep Medicine*, vol. 11, no. 6, pp. 591-592, 2015.
- [2] D. Cary, K. Briffa, and L. McKenna, "Identifying relationships between sleep posture and non-specific spinal symptoms in adults: A scoping review," *Bmj Open*, vol. 9, no. 6, Jun, 2019.
- [3] L. M. Brindles, "Bedsores: What you should know about decubitus ulcers," *Healthline*. Available online: <https://www.healthline.com/health/pressure-ulcer> (December 3, 2021)
- [4] K. Dave, and R. D. Choudhary, "Effectiveness of a pressure ulcer prevention package (pupp) for patients admitted in intensive care units: An experimental study," *International Journal of Advances in Nursing Management*, 2020.
- [5] L. M. Soban, S. Hempel, B. A. Munjas, J. Miles, and L. V. Rubenstein, "Preventing pressure ulcers in hospitals: A systematic review of nurse-focused quality improvement interventions," *Joint Commission journal on quality and patient safety*, vol. 37, no. 6, pp. 245-52, 2011, 2011.
- [6] E. P. Doheny, M. M. Lowery, A. Russell, S. Ryan, and Ieee, "Estimation of respiration rate and sleeping position using a wearable accelerometer," *IEEE Engineering in Medicine and Biology Society Conference Proceedings*, pp. 4668-4671, 2020.
- [7] R. S. Abdulsadig, S. Singh, Z. Patel, and E. Rodriguez-Villegas, "Sleep posture detection using an accelerometer placed on the neck," *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, vol. 2022, pp. 2430-2433, 2022, 2022.
- [8] S. M. Mohammadi, S. Enshaeifar, A. Hilton, D. J. Dijk, and K. Wells, "Transfer learning for clinical sleep pose detection using a single 2d IR camera," *Ieee Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 290-299, 2021.
- [9] A. Y. C. Tam, B. P. H. So, T. T. C. Chan, A. K. Y. Cheung, D. W. C. Wong, and J. C. W. Cheung, "A blanket accommodative sleep posture classification system using an infrared depth camera: A deep learning approach with synthetic augmentation of blanket conditions," *Sensors*, vol. 21, no. 16, Aug, 2021.
- [10] J. E. Kiriazi, S. M. M. Islam, O. Boric-Lubecke, and V. M. Lubecke, "Sleep posture recognition with a dual-frequency cardiopulmonary doppler radar," *Ieee Access*, vol. 9, pp. 36181-36194, 2021.
- [11] Y. C. Yao, L. R. Xu, P. Xia, H. Zhang, L. D. Du, X. X. Chen, Z. Fang, and Ieee, "Highly generalized sleep posture recognition using fmcw radar," *International Conference on Wearable and Implantable Body Sensor Networks*, 2023.
- [12] S. A. Shah, and F. Fioranelli, "RF sensing technologies for assisted daily living in healthcare: A comprehensive review," *Ieee Aerospace and Electronic Systems Magazine*, vol. 34, no. 11, pp. 26-44, Nov, 2019.
- [13] Y. Ge, A. Taha, S. A. Shah, K. Dashtipour, S. Y. Zhu, J. Cooper, Q. H. Abbasi, and M. A. Imran, "Contactless WiFi sensing and monitoring for future healthcare-emerging trends, challenges, and opportunities," *Ieee Reviews in Biomedical Engineering*, vol. 16, pp. 171-191, 2023.
- [14] Q. Q. Sun, E. Gonzalez, Y. Sun, and Ieee, "On bed posture recognition with pressure sensor array system," *IEEE Sensors*, 2016.
- [15] X. W. Xu, F. Lin, A. S. Wang, Y. Hu, M. C. Huang, and W. Y. Xu, "Body-earth mover's distance: A matching-based approach for sleep posture recognition," *Ieee Transactions on Biomedical Circuits and Systems*, vol. 10, no. 5, pp. 1023-1035, Oct, 2016.
- [16] G. Matar, J. M. Lina, and G. Kaddoum, "Artificial neural network for in-bed posture classification using bed-sheet pressure sensors," *Ieee Journal of Biomedical and Health Informatics*, vol. 24, no. 1, pp. 101-110, Jan, 2020.
- [17] A. T. Zhao, J. Y. Dong, and H. Y. Zhou, "Self-supervised learning from multi-sensor data for sleep recognition," *Ieee Access*, vol. 8, pp. 93907-93921, 2020.
- [18] K. Tang, A. Kumar, M. Nadeem, and I. Maaz, "CNN-based smart sleep posture recognition system," *IoT*, vol. 2, no. 1, pp. 119-139, 2021.
- [19] Q. S. Hu, X. C. Tang, and W. Tang, "A real-time patient-specific sleeping posture recognition system using pressure sensitive conductive sheet and transfer learning," *Ieee Sensors Journal*, vol. 21, no. 5, pp. 6869-6879, Mar, 2021.

- [20] H. K. Diao, C. Chen, W. Yuan, A. Amara, T. Tamura, J. H. Fan, L. Meng, X. Y. Liu, and W. Chen, "Deep residual networks for sleep posture recognition with unobtrusive miniature scale smart mat system," *Ieee Transactions on Biomedical Circuits and Systems*, vol. 15, no. 1, pp. 111-121, Feb, 2021.
- [21] H. K. Diao, C. Chen, W. Chen, W. Yuan, A. Amara, and Ieee, "Unobtrusive smart mat system for sleep posture recognition," *IEEE International Symposium on Circuits and Systems*, 2021.
- [22] H. K. Diao, C. Chen, X. Y. Liu, W. Yuan, A. Amara, T. Tamura, B. Lo, J. H. Fan, L. Meng, S. H. Pun, Y. T. Zhang, and W. Chen, "Real-time and cost-effective smart mat system based on frequency channel selection for sleep posture recognition in IoMT," *Ieee Internet of Things Journal*, vol. 9, no. 21, pp. 21421-21431, Nov, 2022.
- [23] Z. Liu, X. Wang, Y. Le and J. Sun, "A study on sleep position recognition of body pressure image based on KPCA and SVM," *IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, pp. 189-193, 2019.
- [24] Z. Liu, X. Wang, M. SU and K. Lu, "A method to recognize sleeping position using an CNN model based on human body pressure image," *IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, pp. 219-224, 2019.
- [25] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, and Ieee, "Rethinking the inception architecture for computer vision," *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 2818-2826, 2016.
- [26] M. Sandler, A. Howard, M. L. Zhu, A. Zhmoginov, L. C. Chen, and Ieee, "MobileNetV2: Inverted residuals and linear bottlenecks," *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 4510-4520, 2018.
- [27] F. Chollet, and Ieee, "Xception: Deep learning with depthwise separable convolutions," *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 1800-1807, 2017.
- [28] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," *Medical Image Computing and Computer-Assisted Intervention*, Pt Iii, vol. 9351, pp. 234-241, 2015.
- [29] K. M. He, X. Y. Zhang, S. Q. Ren, J. Sun, and Ieee, "Deep residual learning for image recognition," *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 770-778, 2016.