DCNN Based Stress Identification on Thermal Images with Neural Pruning Improvement

Yefeng Shen

Research School of Computer Science
The Australian National University (ANU)
Canberra (ACT) 2601 AUSTRALIA
u6226854@anu.edu.au

Abstract. Stress is a common emotion for everyone, often triggered by time pressure, challenges and other factors. People sometimes experience long-term stress without realizing it, which can lead to serious health problems such as nervous tension, emotional irritability, and physical damage. Therefore, it is important to develop an automatically method to help detecting human stress in time. Traditional stress recognition mainly based on the contact between psychological doctors and patients which was inefficient. Later, researchers discovered some symptoms of human stress which can be employed for classification including both physical appearance and physiological response[12]. Recently, many systems have been developed based on extracting features from RGB images and thermal images. However, those approaches are quite complex and time-consuming in defining feature extraction methods and classifiers. Our study aims to automatically extract features from thermal videos, and then efficiently recognize human stress. With a given summarised database including extracted top-5 features from both RGB and thermal videos, we created a 2-layer neural network and optimized its redundant structure with distinctiveness pruning method as the baseline model which achieved 65% prediction accuracy. Then we turned to the source database, ANUStressDB, of 35 thermal videos, and applied deep convolutional neural network(DCNN) based transfer learning(TL) to learn more representative features for stress classification. With a fixed DCNN model, ResNet50, as the feature extractor and a RBF SVM classifier, we achieved a high accuracy of 91%. Furthermore, through finetuning and retraining the whole DCNN models, the performance was further improved to 93%.

Keywords: Thermal Images · Stress Recognition · Transfer Learning · Convolutional Neural Network · Distinctiveness Neural Pruning.

1 Introduction

Scientists have been interested in detecting and identifying human stress for a long time. They found a number of important physical appearance and physiological response to measure such as heartbeat rate, blood flow and so on[12]. Pavlidis et al.[18] were the first to put forward the idea of using a thermal sensor instead of human contact to analysis people behaviours. Since then, many different information extraction methods have been developed and applied to RGB and thermal images for stress identification. Systems with single image source can only achieve a maximum accuracy of 65% while a fusion feature extraction method on both RGB and thermal images combined with a genetic algorithm (GA)-support vector machine (SVM) classifier highly improves the accuracy to 89%. However, it employed manual work to define a template in matching human face and required complex calculation to extract and evaluate the quality of information from images. In contrast, DCNN models have better tolerance of classification tasks and stronger learning ability with diverse filters to extract features automatically. In recent years, DCNN plays an increasingly important role in the field of computer vision, especially for image classification task, which takes the fore-front. In 2012, a new proposed DCNN, AlexNet with 8 layers and the first application of Rectified Linear Units (ReLUs) as activation functions, strongly beat out all competitors and won the ImageNet Large Scale Visual Recognition Challenge[17]. Later, VGG Net demonstrated the benefits of increasing model depth by stacking more convolutional layers and obtained a higher performance than AlexNet in 2014[20]. Considering that our goal was actually to learn and classify human stress based on frames extracted from thermal videos in the ANUStressDB database, this was essentially an image classification problem where DCNN models were expected to perform better. Therefore, I decided to use the transfer learning based on some existing DCNNs to conduct the experiments.

Firstly, we conducted a 2-layer fully connected feed-forward network on the given summarised database which contains top 5 of the features extracted from both RGB and thermal videos in the ANUStressDB database. Since it applied the the same complex extraction method as in [12], we used these features to test a baseline accuracy to compare the performance of DCNN models. The structure of the simple neural network classifier was optimized by using the distinctiveness neural pruning method which efficiently removed redundant elements inside[7]. The baseline accuracy was about 65%. Then, we moved on to the experiments based on DCNN models. There were two main applications of transfer learning included. At the first stage, we applied the ResNet50 with fixed weights to automatically extract high-level features from thermal images, then feeding the feature map to classifiers such

as diverse SVM and simple neural networks optimized by distinctiveness neural pruning method to compare their performance. From that, we obtained a high accuracy of 90%. In the second stage of the experiments, we fully adopted the deep CNN models for training and prediction including VGG16 and ResNet50 models. We first finetuned the layer structures and parameters to fit our database condition, and then trained the whole network to update the weight. Based on that, the testing accuracy was further improved to 93%. All experiments were conducted on a 2.6 GHz Intel Core i7-10750H CPU with 16 GB of RAM and a NVIDIA GeForce RTX 2070 GPU with 16 GB of memory. The work was implemented in Jupyter Notebook, Python, and other supporting libraries such as Sklearn and Pytorch.

2 Method

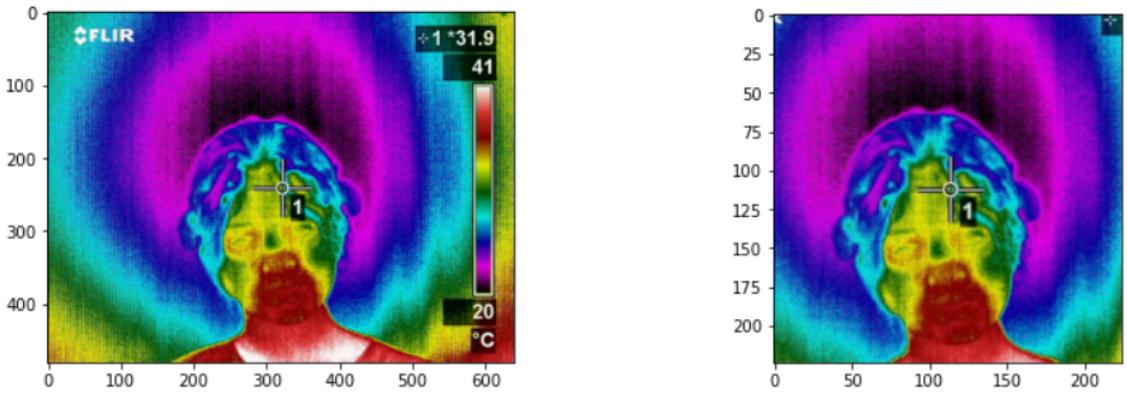
2.1 Data Pre-processing Method

There are two databases included in this paper. One is a summarized database including 620 observations with 5 most representative features extracted from both RGB and thermal videos. Each observation is labeled as stressful or calm. The summarised database already contains useful features but have different scales and distributions which is difficult to learn a stable model from them. Normalising features into a standard range can reduce the computational burden of large values[14] and speed up the training process to convergence[15]. Therefore, we converted features to float type and then applied the standard score(Z-score) normalization on every feature column. The equation to calculate Z-score is shown below:

$$Z = \frac{x - \mu}{\sigma} \tag{1}$$

where x is one observation value, μ is the mean of a feature column and σ is the standard deviation. Besides, The target label column is converted to one and zero for the convenience of building the predictive model.

Another one is 35 thermal videos collected by researchers from Australian National University(ANU) with an experiment. There were 35 participants in total including 22 males and 13 females, aged between 23 and 39. The instructor played a short film made from a collection of 20 negative or positive clips as a stress motivator and recorded videos with a camera set up to catch subjects' face. Each clip last about 1.5 minutes, with 5 seconds between them to calm down the participants. Videos were labeled according to the ground truth of the sequence of clips in the film as stressful or calm. Therefore, each video had 20 labels. There were two sets of labeled videos with their filename start with TCS or TSC. They only differ in the first six labels while TCS videos are start at 3×calm then 3×stressful and TSC are 3×stressful then 3×calm. When I looked into the thermal videos in ANUstressDB, I found there is 5 to 8 seconds setup at the beginning with the instructor moving around in the background. So I prepared to extract frames from the 9th second. During the 90 seconds of each clip, the label of each frame was assumed to be the same as the ground truth. The gap between each segment is 5 seconds long. Based on these rules, I took one frame every 10 seconds from the first 6 clips of each video which lead to a total of 60 images. After processing the whole 35 videos, I got 2100 labeled images with the resolution of 640×480. As you can see from the original extracted frame in Figure 1a, there are some texts and a color bar around the image. However, identifying the emotion of people only requires information learned from the center thermal human face. To remove the noises, all image were first rescaled to the dimension of 256×256, and then a 224×224 square area were decided to be cropped right in the center of images as you can see in Figure 1b. Because the camera was positioned directly in front of the subjects, most of the subjects' faces were in the center of the scene. Considering that we were going



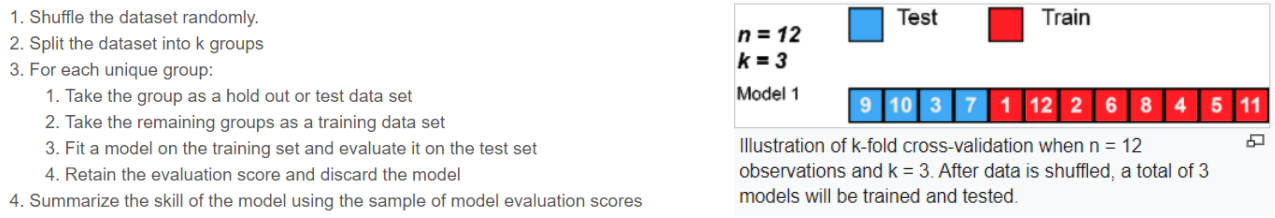
(a) The original extracted frame(640×480). (b) Rescaled and cropped image(224×224).

Fig. 1: Image pre-processing

to apply pre-trained DCNN based transfer learning, we calculated the channel-wise Z-score again to normalize our

224×224 images with the same mean [0.485, 0.456, 0.406] and standard deviation[0.229, 0.224, 0.225] computed on ImageNet where the DCNN models trained on.

For the purpose of building a predictive model, our main goal is to achieve the possible best accurate predictions on not only the user data but also first seen data of the same type. So the evaluation of models is of great significance. Although the extracted image database was balanced with 1050 stressful samples and 1050 calm samples, bias would happen in sub sets after a random splitting method. Besides, training and evaluating on certain set is not reliable enough because there may exist potential problems like over-fitting or bias in split subsets[3]. Compared with that, we implemented a 5-fold cross validation which would allow models to be trained and tested 5 times on different subsets of all observations. The averaged accuracy of all folds were recorded which can better reflect the generalization ability and prediction level of the constructed models. To further eliminating the bias in sub-samples, this paper conducted the stratified 5-fold cross-validation which ensures that the proportions of the two target classes stay roughly the same in each partition. The general procedure and sample display image can be found in Fig. 2.



(a) The general procedure of k-fold Cross Validation

(b) Sample image of 3-fold cross validation

Fig. 2: (a): Image extracted from the website Machine Learning Mastery[13]; (b): Image extracted from Wikipedia[5].

2.2 Simple baseline classifier – 2-layer fully connected feed-forward network

The topology of the initial network was 10-10-2, being ten input features, ten hidden neurons, and two output neurons. The target column in the data set contains only two classes, stressful and calm, which lead to a binary classification problem. The model was a simple fully connected network without lateral, recursion or convolution. We have applied the basic sigmoid function as the activation of the hidden layer and the output layer to predict values between 0 and 1. Considering our classification task, the loss function in backpropagation was decided to be cross entropy loss. The initial optimiser was set to be Adam with learning rate equals 3e-4. The default epochs is set to be 1000 times. After each epoch, the model would predict on the validation data set and record the relevant accuracy. Only the model of the best validation accuracy would be returned and then tested the performance on the testing set.

Distinctiveness Pruning The first coming problem of pruning is to decide the appropriate pruning time. We prefer to prune neurons after the training converges, so the pruning is set to be activated after 300 epochs to guarantee the convergence. Then, we should read in the activation output of weight matrix first and normalize its value by minus 0.5. Each row vector of the activation matrix represent one neuron. According to Gedeon's research[7], angle between each two vectors would be computed. Angle lower than 15 degrees indicates high similarity, so one of the neuron should be removed and the associated weights should be added to the neuron kept. Angle larger than 165 degrees means they are complementary pairs which can be removed at the same time[9]. A mask of one and zero was created to block the removed neurons that is all weights of a removed neurons would become zero at every epoch. This paper performs two approches to compute angles. One is mentioned in Gedeon's paper with the equation shown in (3). For convenience, this method would be referred as ArcTan Distinctiveness Pruning(ATDP).

$$angle(i,j) = \arctan(\sqrt{\frac{i^2 \times j^2}{(i \times j)^2}} - 1) \tag{2}$$

Another is the pure mathematical computation of cosine similarity between vectors as shown in (4). For convenience, it would be referred to as Cosine Distinctiveness Pruning(CDP) method.

$$angle(i,j) = \arccos(\frac{i \cdot j}{|i| \times |j|}) \tag{3}$$

where i and j are normalised activation vectors.

2.3 Deep CNN model based Transfer Learning

I determined to employ some state-of-the-art deep convolutional neural network(DCNN) models on our dataset instead of defining and implementing complex feature extractors and classifiers. Especially since PyTorch provides a large variety of DCNN models that have been trained and saved on sufficient data from ImageNet including 1.2 million images from 1000 categories [6], it is very convenient to implement transfer leaning(TL) with these pre-trained models nowadays rather than training entire DCNN models from scratch (with random initialization) which can not only guarantee the performance of DCNN models on our image dataset but also greatly improve the efficiency of training. Our experiments included the two major applications of TL.

Fixed Feature Extractor One is utilising the pre-trained network as a fixed feature extractor. Without training to update weights, the base convolutional network can already learn meaningful features quickly. I conducted the ResNet50 model to learn features from my dataset which was widely used in TL. The average pooling layer of ResNet50 was selected as the end layer which would return a vector of 2048 features for each input observation. After processing all images in the dataset, a large feature map with dimension of 2048 would be prepared for running different classifiers. I mainly experimented SVM classifiers and a 2-layer fully connected feed-forward network optimized by distinctiveness neural pruning method as classifiers.

Support vector machine(SVM) classifer – SVM is a supervised ML algorithm classifying classes by finding the hyperplane that maximizes the margin from each class which has been widely employed in the area of stress recognition[12]. SVM requires complex data transformation to raise dimensions but can be sidestepped by using some kernel functions to replace the expensive dot product in computations of the new dimensions. There are several common kernel functions available such as linear, polynomial, and RBF. Thus, SVM is effective in dealing with high dimensional spaces which guarantees its performance in dealing with the feature maps learned by DCNN model. In nonlinear classification, RBF often performs the best among all kernel functions[2]. Therefore, I trained the SVM classifer with the Radial Basis Function(RBF) kernel: $exp(-\gamma\|x - x'\|^2)$, where γ determines the effect of a single training input. Another important parameter in defining RBF is C which determines the smoothness of decision surface. I implemented experiments on the value of input parameters gamma and C with choices of [0.5, 0.01, 1e-3, 1e-4] and [1, 10, 100]. Only the SVM with the best performance was recorded.

Finetuned DCNN based Image Classification The second way to apply TL is fine-tuning the pre-trained models. It is necessary to reset the size of final fully connected layer since the number of target classes in ImageNet is 1000 but is only 2 in our dataset(stressful and calm). Then, we should unfreeze the weights of all layers to be updated during training so that the generated models can fit better with our data condition. The selected DCNN models include VGG16 and ResNet50.

VGGNet as mentioned in the section, Instruction, has presented great performance in image classification with deep structure by stacking convolutional layers. However, along with the increase of the depth of neural networks, a potential risk of too many parameters began to surface which was challenging for device to handle very deep convolutional neural network. Besides, the accuracy often get saturated, even degrading rapidly which illustrates that deep models may not be optimized well[11]. Residual learning was then introduced with the new ResNet model by He et al. in 2016. The main structure, residual learning block, of ResNet is shown in Fig. 3. Compared with

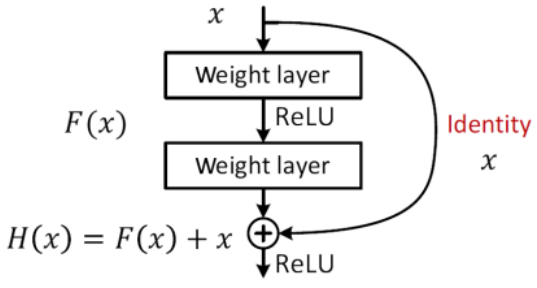


Fig. 3: Residual Learning Block

traditional neural network, it uses a skip connection(identify connection) between layers to combine different level of features to improve the gradient propagation process so that the vanishing gradient and degradation problem of deep neural networks can be eliminated. ResNet achieved the 1st place performance in completing the ILSVRC 2015 image classification task[10]. ResNets with different depth were implemented and expected to achieve the highest stress prediction accuracy on my dataset.

3 Results and Discussion

3.1 Evaluation on Baseline Model

For the baseline model, I implemented both ATDP and CDP angle calculations to remove neurons. Considering that larger hidden layer size would lead to higher probability of similar or complementary neurons, I tested different number of hidden neurons from 20 to 5. Besides, an pruning ratio is computed under the best model to reflect the effects of Distinctiveness Pruning as well. During the experiments, the standard degree bounds[9] to remove neurons was found to work poorly on the stress recognition data set. In order to successfully test both of these methods in practice, I decided to exaggerate the deletion range that is removing neurons with angle lower than 30 or greater than 150. Only the optimal model of using ATDP and CDP would be recorded. By further analysing the Table 1, it is clear that CDP performed much better than ATDP. models with CDP achieved the highest testing accuracy of 65.184% with a 17% pruning ratio. The low pruning ratio and small increase in accuracy indicated that ATDP could not detect and remove neurons effectively on the given data set.

Pruning Method	Layer size	Angle calculation	Degrees to prune [Lower bound, Upper bound]	Average Pruning ratio	Average Testing Accuracy of 5 tests Run
No Pruning	5	None	None	0%	54.012%
Distinctiveness Pruning	20	ATDP	[50,130]	4%	58.52%
Distinctiveness Pruning	20	CDP	[30,150]	17.0%	65.184%

Table 1: Evaluation on the function of distinctiveness pruning method with CDP and ATDP.

3.2 Evaluation on TL Based Stress Recognition

I implemented experiments on diverse SVM classifiers with different kernel functions and recorded the one with highest testing accuracy. As for fine-turned DCNN models, the default loss function was set to be cross entropy for classification task and the default batch size was 32. I determined to adopt stochastic gradient descent(SGD) optimizer with a start learning rate of 1e-3 and a momentum of 0.9 for each model. The learning rate(LR) was set to be decreased by a LR scheduler which would multiply the LR with a gamma coefficient of 0.1 after every 7 epochs. With some early experiments, I found 10 epochs was enough for DCNN models to converge on the image database. The default splitting method was 9-fold cross validation as used in most model testing experiments. During training in each fold, I would computed the the loss and accuracy on both training and validation set after each epoch. The best weights of a model would be updated and saved only if a higher accuracy appeared on the validation set. The performance was evaluated by calculating the accuracy(ACC), Precision(PPV), Recall(TPR), and F1-score based on the confusion matrix of testing set. The testing results in table 2 and table 4 would only show the average value of 9 folds. Obviously, DCNN models had a great performance in thermal image based stress recognition. Even

Model	ACC	PPV	TPR	F1-score
ResNet50 feature Extractor + RBF SVM classifier	91.43%	92%	91%	91%
ResNet50	92.48%	92.38%	92.57%	92.471%
ResNet18	92.76%	93.09%	92.57%	92.76%
VGG16	92.76%	93.49%	92.00%	92.71%
ResNeSt50	93.05%	92.89%	93.33%	93.08%

Table 2: Evaluation on diverse DCNN based models

the SVM classifier with a fixed pre-trained ResNet50 model as the feature extractor achieved a surprisingly high accuracy of 91.43%. I then firstly implemented the same ResNet50 model but finetuned its structure for retraining. The result was also very high but had a potential risk of over-fitting because the training accuracy achieved 100%. Therefore, I decided to simplify its structure by testing the ResNet18 model with lower depth which turned out to perform better than ResNet50 with a higher accuracy of 92.76%. Besides, I also experimented a simpler VGG16 model which obtained the same accuracy as ResNet18. Furthermore, an advanced ResNet model, ResNeSt, was also conducted. This model applied a split attention method similar to human visual attention to pay more attention to

informative region of images and ignore useless background. Compared with ResNet50, ResNeSt50 with the same depth showed better prediction accuracy, reaching the highest 93.05%. The Figure 4 presented the confusion matrix predicted by the worst and best model in Table 2. Compared with the SVM classifier, we could clearly notice that ResNeSt correctly classified 9 stressful images from the wrong calm class in the right image of Figure 4.

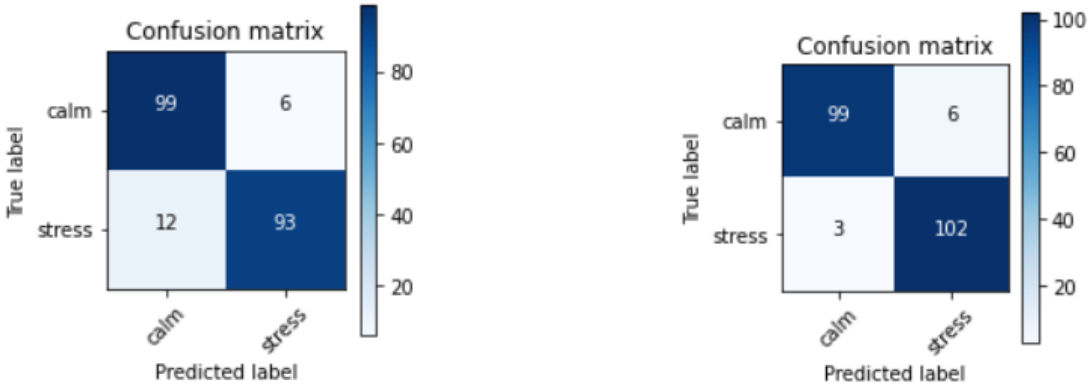


Fig. 4: The result confusion matrix predicted by SVM(left) and ResNeSt50(right)

3.3 Evaluation on State-of-the-art Classifiers

In the Table 3, we compared our baseline model and optimal DCNN model with 5 other classifiers[19, 12] on the same original stress database, ANUstressDB. They applied LBP and temerature of super pixels to extract features from the source RGB and thermal modalities and then implemented some special SVM classifiers. Although we

Models	Testing Accuracy
(V_{LBP} + T_{LBP}) with SVM classifier[19]	61%
2-layer NN with Distinctiveness Pruning	65%
(V_{LBP} + T_{LBP}) with Genetic Algorithm SVM classifier(GASVM)[19]	79%
(V_{LBP} + T_{HDTP}) with SVM classifier[19]	76%
(V_{LBP} + T_{HDTP}) with Genetic Algorithm SVM classifier(GASVM)[19]	85%
Modality Fusion[12]	89%
ResNet50 feature Extractor with RBF SVM classifier	91.43%
ResNeSt50	93.05%

Table 3: Comparison between the optimal NN model and state-of-the-art classifiers.

only work on the thermal videos with less information, the accuracy of our optimal ResNeSt based stress classifier beat all previous complex classifiers. Compared with them, DCNN models are not only easy to implement but also surpass all the others in stress recognition with the highest accuracy of 93.05%.

4 Conclusion and Future Work

Overall. We successfully carried out the distinctiveness pruning with both angle calculations CDP and ATDP and proved their positive effect in improving the structure of neural networks. During the training, we also found that CDP performs better than ATDP in detecting and removing redundant neurons. Most importantly, by applying DCNN based transfer learning, we beat all the previous stress recognition systems built on the ANUSressDB database. And we only used thermal videos, a better predictive performance than 93% would be expected if we used the full database including RGB videos. As for future tasks, I would think about applying the distinctiveness pruning to improve the structure of some layers in DCNN and experimented more existing state-of-the-art DCNN models to find better stress classifier.

References

1. Blalock, D., Ortiz, G., Frankle, J., Guttag, J.: What is the State of Neural Network Pruning? arXiv preprint arXiv:2003.03033 (2020)

2. B. Divya and M. Santhi: SVM-based Pest Classification in Agriculture Field. International Journal of Recent Technology and Engineering (IJRTE) **7**(5S3), 2277–3878 (2019)
3. Cawley, G. C., Talbot, N. L.: On over-fitting in model selection and subsequent selection bias in performance evaluation **11**, 2079–2107 (2010)
4. Ciresan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2012, pp. 3642–3649. https://doi.org/10.1109/cvpr.2012.6248110
5. Cross validation (statistics), https://en.wikipedia.org/wiki/Cross-validation (statistics)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255. Ieee (2009)
7. Gedeon, T.D., Harris, D.: "Network Reduction Techniques," PROCEEDINGS INTERNATIONAL CONFERENCE ON NEURAL NETWORKS METHODOLOGIES AND APPLICATIONS 1991, AMSE, vol. 1, pp. 119-126. San Diego (1991).
8. Gedeon, T.D.: Data mining of inputs: analysing magnitude and functional measures. International Journal of Neural Systemsc **8**(02), 209–218 (1997)
9. Guanjie H.: Distinctiveness Pruning on 'fight or flight'Response Prediction LSTM Network.
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778. (2016)
11. Raimi K.: Illustrated: 10 CNN Architectures Towards Data Science, https://towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d.
12. Irani, R., Nasrollahi, K., Dhall, A., Moeslund, T.B. and Gedeon, T.: Thermal super-pixels for bimodal stress recognition. In 2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA), pp. 1–6. IEEE. (2016)
13. Jason B.: A Gentle Introduction to k-fold Cross-Validation, https://machinelearningmastery.com/k-fold-cross-validation/ (2018)
14. Jason B.: How to use Data Scaling Improve Deep Learning Model Stability and Performance, https://machinelearningmastery.com/how-to-improve-neural-network-stability-and-modeling-performance-with-data-scaling/ (2019)
15. Jeremy J.: Normalizing your data (specifically, input and batch normalization), https://www.jeremyjordan.me/batch-normalization/ (2018)
16. Keskar, N.S., Socher, R.: Improving generalization performance by switching from adam to sgd. arXiv preprint arXiv:1712.07628 (2017)
17. Krizhevsky, A., Sutskever, I. and Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25**, 1097–1105 (2012)
18. Pavlidis, I., Levine, J.: Thermal image analysis for polygraph testing. IEEE Engineering in Medicine and Biology Magazine **21**(6), 56–64 (2002)
19. Sharma, N., Dhall, A., Gedeon, T., Goecke, R.: Thermal spatio-temporal data for stress recognition. EURASIP Journal on Image and Video Processing, **1**, 1–12. (2014)
20. Simonyan, K. and Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)