# Dynamic Noise Injection for Facial Expression Recognition In-the-Wild

SangHwa Hong[1] and Jin-Woo Jeong[12*]
Department of Industrial Engineering[1], Department of Data Science[2]
Seoul National University of Science and Technology
Gongreung-ro 232, Nowon-gu, Seoul, South Korea
{hongsw5911, jinw.jeong}@seoultech.ac.kr

## Abstract

*Facial expression-based emotion analysis is one of the most important artificial intelligence research fields. However, a lot of works still suffer from the low classification/regression performance caused by overfitting. Therefore, we propose new noise injection techniques to alleviate the overfitting problem on the task of facial expression recognition in the wild. Specifically, both techniques are based on the ResNet-18 architecture, and we periodically or dynamically add feature-level noise into the BN+ReLU unit to learn more robust features. The periodic method needs to probe the optimal hyperparameter with respect to the interval for the noise injection through trials and errors. Therefore, we propose the second method in order to make a dynamic noise injection mechanism work without a non-trivial time-consuming hyperparameter search process. Finally, the performance of the two methods is reported in the experiment. Our experiments on facial expression classification with the AffectNet dataset demonstrated the usefulness of the proposed approach.*

## 1. Introduction

Facial Expression Recognition(FER) is an important problem in emotion analysis research through artificial intelligence. As the human facial expression is one of the most natural and efficient ways to express human emotions, it is expected that AI-based agents would be able to identify and imitate human emotions from facial expressions during communication and interaction with users in the near future [18]. In addition, facial expression recognition can be applied to various industrial domains, such as health care and entertainment, to improve user experience [19].

A lot of studies have shown that it is possible to perform facial expression-based human emotion analysis using CNN, one of the deep learning technologies, with higher performance than traditional machine learning methods [18]. However, most of the previous studies [4,13] have worked on FER using facial images captured under controlled environments (e.g., lab study) [16, 23] with discrete emotional stimulation. Therefore, such methodologies cannot be well applied to practical applications since they are not able to handle facial expressions captured in the wild. Moreover, since deep learning models tend to remember all information in the data [21], an overfitting problem, can be much more severe in the practical FER domain [9, 15] because emotion analysis through facial expression recognition still suffers from the lack of large-scale in-the-wild data.

In order to solve the aforementioned problem, a solution that can distinguish the representative features of the data from the noise is required [21]. Typical solutions for this include reducing model complexity, ensemble, and regularization. There exist various techniques for regularization, and representative techniques include parameter norm penalty, dataset augmentation, early stopping, and noise injection [2]. Of these techniques, noise injection is known to be effective for circumventing overfitting and enhancing generalization in deep learning [6]. There have been a lot of attempts for noise injection, which can be categorized by 1) when the noise is injected (e.g., train, test phase), or 2) where the noise is injected (e.g., input, hidden, and an output layer of neural networks). For instance, Gaussian noise [3] injection adds noise into the hidden layer during the training phase, while label smoothing [20] the noise into the output layer during the training phase.

In general, most noise injection methods usually have attempted to add small noise continuously in the training

*Corresponding author

phase for regularization. However, these methods may not improve performance due to the nature of the domain [1]. Unlike traditional noise injection techniques in which noise injection is continuously performed, in this paper, we propose a novel approach to dynamically inject noise during a training process. Specifically, we inject feature-level noise into the layer after batch normalization in the network for several selected epochs during the training phase. For this, we propose two methods to determine when the feature-level noise is injected into the network: 1) periodic method and 2) dynamic method. In the first approach, a model is given the feature-level noise every $T$ epochs. Conversely, the second approach automatically determines the timing for noise injection through the use of a modified ReLU function. Through these methods, the model can be trained to distinguish the representative features of the data from the noise well. Finally, the experiment on facial expression classification in the wild using the AffectNet database revealed that the proposed approach could contribute to the higher generalizability of the FER model in the wild.

## 2. Overfitting and Regularization

Overfitting is one of the widely recognized issues in deep learning. There are various causes, but representative ones include model complexity [22], a small amount of data, and a large amount of data noise. This problem is never applicable in real applications as it produces results that cannot be generalized to unseen data during training. To solve this problem, many studies have been conducted in terms of regularization to construct a generalizable model. Regularization is a popular technique used to prevent overfitting and improve the generalization performance of a neural network model. Generally, regularization methods add a penalty term to the loss function of a model or injects some noise to the layers to discourage the model from overfitting the training data [11]. For example, L1 regularization, L2 regularization, Dropout, Data augmentation, Early stopping, Batch normalization, DropConnect, and noise injection are the most commonly used methods.

Dropout is one of the popular regularization methods that involves randomly dropping out or deactivating some neurons during training, which means that several neurons do not contribute to the forward or backward propagation of the network for a particular training example. Every time a dropout mask is applied during training, a different set of neurons is dropped out, resulting in a different neural network with different activations. Therefore, it is often said that the use of dropout during training effectively creates an ensemble of multiple neural networks with shared weights [7].

On the other hand, noise injection techniques add noise data to improve the generalization performance of neural network models. The goal of this technique is to improve the signal-to-noise ratio (SNR) by increasing the strength of the signal relative to the noise. It is also known that noise injection is more effective when noise not correlated with representative features of the data is given [5, 15]. By adding the noise to the input, hidden layer, parameters, and output layer, the model is forced to learn more robust features that are less sensitive to variations in the input data [12]. This leads to the improvement of the generalization performance of a deep learning model. Generally, it involves continuously adding random small noises to the input or hidden layers of the model during training.

It is obvious that previous studies on regularization have contributed to performance improvement in various domains which have enough training and validation samples. However, FER in-the-wild domain in particular still suffers from the lack of large-scale data as well as an overfitting problem, the effects of existing regulation techniques might not be working well. Therefore, we explore how popular regulation techniques, such as dropout and noise injection work for this domain, and propose an improved version of noise injection to improve the robustness of a deep learning-based FER model.

## 3. Method

In this section, we describe the proposed dynamic noise injection method in more detail. First, we explain the period-based noise injection method which adds a feature-level noise for every fixed interval. Then, we describe an improved version of the dynamic injection method which automatically determines the time for adding noises.

The model used in this study is Resnet18, which consists of a sequence of Conv+BN+ReLU modules. In the proposed approach, we modified the BN+ReLU module to simply change the sign of the feature value for certain epochs, resulting in noisy input data for the subsequent layers. This provides a similar effect as if large noises are given to the network when the model is utilized in a practical environment, different from the small noises used in existing noise injection techniques.

### 3.1. Periodic Noise Injection

The workflow of the proposed period-based noise injection is shown in Algorithm 1. In this approach, the

**Algorithm 1:** Period-based noise injection

```
interval = T
step = 0
flag = False
while network training do
    step += 1
    if flag equals to True then
        After Batch Normalization:
        feature *= -1 (noise injection)
    end
    if step % interval == interval - 1 then
        flag = NOT flag (toggle)
    end
end
```

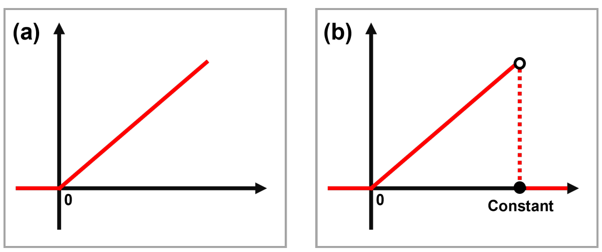


Figure 1. Original ReLU (a) and CliffReLU (b)

interval $T$ is a hyper-parameter used to determine the timing for toggling the flag, therefore, it is necessary to manually find and set an appropriate value. Through various experiments, we found that 1) if the time interval is set too short, a model tends to be not trained; 2) with too long intervals, though training occurred, the model performance did not show much improvement. Therefore, the interval parameter should be carefully selected to achieve performance improvement in FER models, which is not trivial. To overcome this limitation, we also propose a method to automatically add the noise to the layer through a modified version of the ReLU activation function.

## 3.2. Dynamic Noise Injection

In period-based noise injection, the interval $T$ has to be manually set to change the sign of feature maps after the BN+ReLU unit. In order to determine the optimal interval value for the periodic noise injection, a lot of trial and error is required. To solve this issue we proposed dynamic noise injection by using AutoReLU also introduced in this paper. Dynamic noise injection method automatically changes the sign of feature values through a modified version of ReLU activation called AutoReLU1.

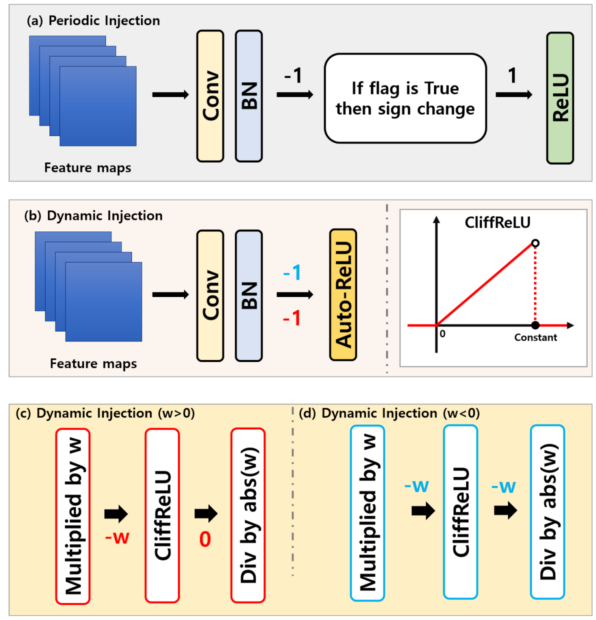


Figure 2. Process of the proposed noise injection methods

### 3.2.1 AutoReLU

As shown in Fig 2 (c) and (d), AutoReLU has three stages. First, we compute element-wise product of the feature representations from previous layers by weight parameter $w$. Second, these features go through CliffReLU function, which drops the value over Constant into zero, also represented as follows :

$$CliffReLU\begin{cases} y = x & (0 < x < Constant) \\ y = 0 & (x < 0, x > Constant) \end{cases} \tag{1}$$

where $y$ denotes activated feature representation after CliffReLU and $x$ is input. *Constant* is arbitrary value which is hyperparameter. Activated features are divided by absolute value of $w$. Note that AutoReLU can be applied only if all $w$ are initialized with a positive value, but less than 1. Finally, AutoReLU is illustrated by equation 2.

$$y = CliffReLU(xw)/w \tag{2}$$

With implemented AutoReLU, dynamic noise injection is conducted as follows:

- Step 1: As learning progresses, all $w$ converges to a value close to 0 and most of $w$ remains in the positive value.
- Step 2: At a certain forward step, all $w$ changes to a negative number close to 0, which is caused by the

| | Resnet18 | +Dropout | +Periodic injection | +Dynamic injection |
|---|---|---|---|---|
| Train Loss | 41.24 | 54.99 | 37.56 | 36.91 |
| Validation F1 | 0.34 | 0.32 | 0.38 | 0.39 |

Table 1. Summary of performance evaluation

re-initialization of each $w$ into a mean of all $w$ before each forward step starts, and the sign remains negative for a certain period of time.

- Step 3: This process repeats changing values of w from positive to negative, and vice versa.

As shown in Figure 2, if $w$ is a positive number as in the red part(Fig 2(c)), it proceeds in a very similar way to the process of period-based noise injection with a False flag(Fig 2(a)), which means we do not add any noise. However, when all $w$ is changed to a negative number as in the blue part(Fig 2(d)), the noise injection is applied.

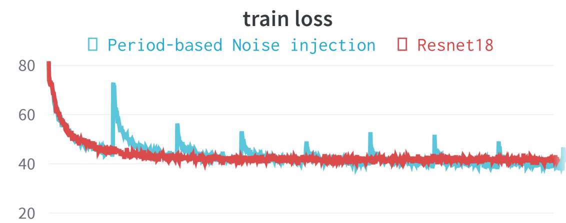


Figure 3. Training loss of Resnet18 and Resnet18 with period-based noise injection

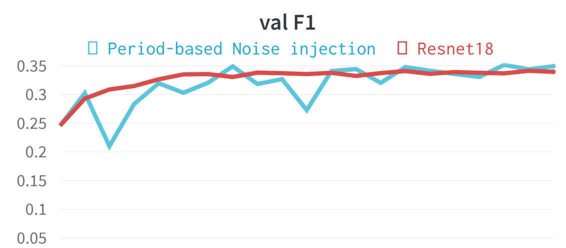


Figure 4. Validation F1 score of Resnet18 and Resnet18 with period-based noise injection

## 4. Experiments and Results

### 4.1. Dataset

We evaluated the effectiveness of the proposed method for the facial expression classification task with the AffectNet [14] dataset. The dataset was created to enable the development and evaluation of facial expression recognition algorithms in real-world scenarios. The images included in the dataset are divided into seven basic emotions: anger, disgust, fear, happiness, sadness, surprise, and neutral. The AffectNet dataset has been widely used in research on facial expression recognition, emotion detection, and affective computing. It has also been used in various applications, such as human-computer interaction, virtual reality, and robotics. Through various experiments, we report the F1 score from the validation phase and training loss.

### 4.2. Training detail

We set our training environment as follows. A batch size of 64 and the ADAM optimizer with a learning rate of 0.0001, and weight decay with a value of 0.0001. All models were trained from scratch.

All the experiments were conducted using a GPU server equipped with two NVIDIA RTX 3090 GPUs, 128 GB RAM, and an Intel i9-10940X CPU. We used the Pytorch framework for the implementation, training, and evaluation of the model.

### 4.3. Baseline versus period-based noise injection

To evaluate the effectiveness of the proposed periodic noise injection approach, we compared the classification performance of the original ResNet18 (baseline) and that of the Resnet18 to which the period-based noise injection technique is applied. As shown in the table1, in the case of the baseline, the max F1 score was observed to be about 0.34, whereas, the period-based noise injection technique achieved an F1 score of 0.38. In addition, train loss of ResNet18 with periodic noise injection is observed to be lower than the baseline by 3.68. Such result can be explained by the following. The most notable feature of the periodic dynamic injection method is that the training loss increases explosively at a specific time point and the validation score for the F1 score dropped drastically, which is caused by noises injected periodically. This trend can be shown in Fig. 3, 4.

In addition, ResNet's building blocks consist of three BN+ReLU units. Through various experiments, we could find that adding noise into only a single BN+ReLU unit in the block led to the best performance. There was no significant difference in performance depending on where the noise was injected. Conversely, if noise was injected into all of the BN+ReLU units together, we could not find any performance improvement.

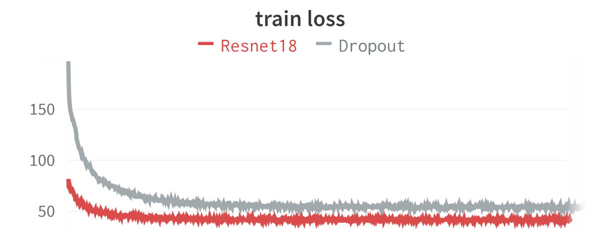


Figure 5. Training loss of Resnet18 and Resnet18 with dropout

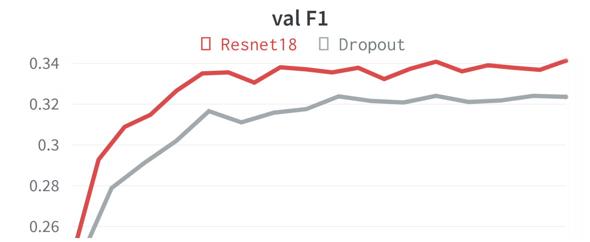


Figure 6. Validation F1 score of Resnet18 and Resnet18 with dropout

## 4.4. Effect of Dropout to the Baseline

It is well known that adding a dropout module to the convolution layer improves the performance of CNNs for various classification tasks [17]. Therefore, we applied dropout to the baseline model and evaluated its performance in facial expression classification in the wild. However, as shown in Table 1, it was observed that the performance of ResNet18 with dropout was even lower than the original ResNet18 by 0.02. In other words, it could be said that the dropout technique might not be suitable for the FER in-the-wild domain. However, we could observe that the periodic noise injection method could improve the performance even when it is applied to baseline with dropout (See Fig. 5, 6,9).

In the case of a general-domain datasets, such as CIFAR-10 [8] and Stanford Cars [10], the performance improvement was observed in Resnet18 with dropout applied. In other words, the dropout technique is not suitable for the expression-based emotion recognition domain. However, it was confirmed that the period-based noise injection method improves the performance (i.e., 0.35 in terms of F1 score) even when it is applied to the baseline with dropout (See Fig. 7, 8).

## 4.5. Periodic vs Dynamic noise injection

Finally, we show the effectiveness of the proposed dynamic noise injection method. In Table 1, we can see the training loss and F1 scores of the Periodic injection

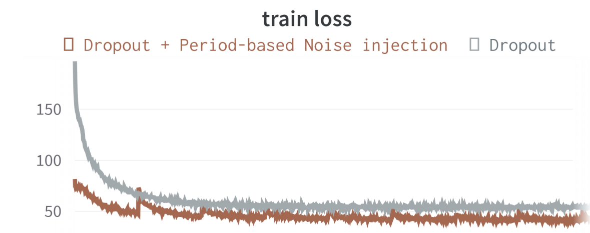


Figure 7. Training loss of Resnet18 with dropout and Resnet18 with both dropout and period-based noise injection

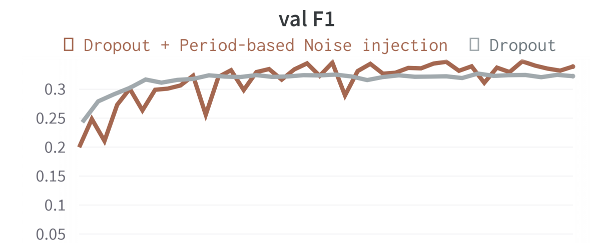


Figure 8. Validation F1 score of Resnet18 with dropout and Resnet18 with both dropout and period-based noise injection

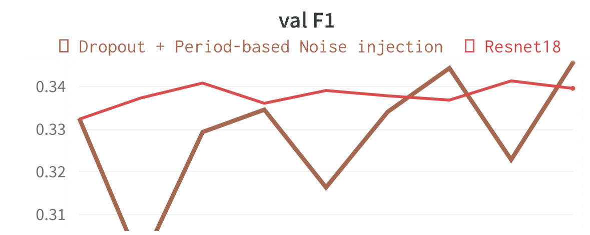


Figure 9. Validation F1 score of Resnet18 and Resnet18 with both dropout and period-based noise injection

and Dynamic noise injection, and it turns out that there is no significant difference between them. It shows that there is no need to undergo the process of trial and error in order to find the optimal interval $T$ (Refer to Fig. 10, 11). In particular, through the experiments, we could find that dynamic noise injection more frequently and irregularly injects feature-level noise into the network compared to the periodic approach, while achieving a similar level of performance.

## 5. Conclusion

Facial expression-based emotion recognition in the wild still suffers from the low classification performance caused by overfitting problems. Therefore, the use of appropriate regularization techniques is very important to improve the robustness of models. In this paper, we

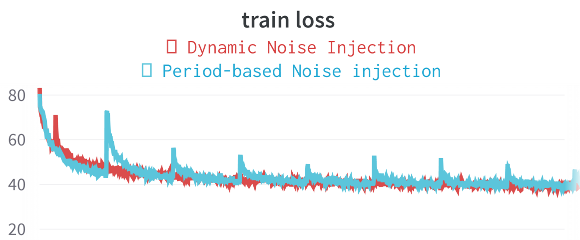


Figure 10. Training loss of Resnet18 with period-based noise injection and Resnet18 with dynamic noise injection

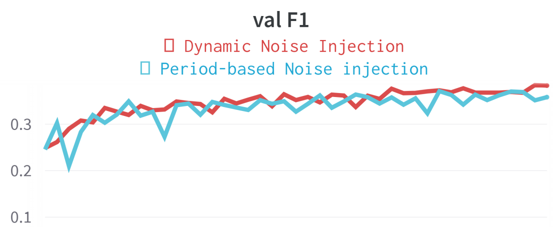


Figure 11. Validation F1 score of Resnet18 with period-based noise injection and Resnet18 with dynamic noise injection

proposed a new type of noise injection technique that can adaptively and dynamically add feature-level noise to the network during the training process. Through various experiments, it was confirmed that the proposed technique contributed significantly to performance improvement for the facial expression classification task in the wild. However, we still have several limitations that need to be addressed in the future: 1) the proposed techniques are tested only for the Resnet18 model. More recent networks need to be tested as well. 2) currently we only evaluated our approach on the AffectNet dataset. Our future study will include more datasets and networks as well as different kinds of tasks (e.g., VA estimation, AU detection, etc). 3) we could observe that a learning rate scheduler should be carefully used in the learning process because weights are not updated after a certain epoch in some cases.

## 6. Acknowledgement

This work was supported by the NRF grant funded by the Korea Government (MSIT)(2023R1A2C1006300)

## References

[1] Houda Abouzid, Otman Chakkor, Oscar Gabriel Reyes, and Sebastian Ventura. Signal speech reconstruction and noise removal using convolutional denoising audioencoders with neural deep learning. *Analog Integrated Circuits and Signal Processing*, 100:501–512, 2019. 2

[2] Mohammad Mahdi Bejani and Mehdi Ghatee. A systematic review on overfitting control in shallow and deep neural networks. *Artificial Intelligence Review*, pages 1–48, 2021. 1

[3] Alexander Camuto, Matthew Willetts, Umut Simsekli, Stephen J Roberts, and Chris C Holmes. Explicit regularisation in gaussian noise injections. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16603–16614. Curran Associates, Inc., 2020. 1

[4] Fei Cheng, Jiangsheng Yu, and Huilin Xiong. Facial expression recognition in jaffe dataset based on gaussian process classification. *IEEE Transactions on Neural Networks*, 21(10):1685–1690, 2010. 1

[5] Wei Cui and Wei Yu. Uncertainty injection: A deep learning method for robust optimization. *IEEE Transactions on Wireless Communications*, 2023. 2

[6] Ruili Feng, Deli Zhao, and Zheng-Jun Zha. Understanding noise injection in gans. In *International Conference on Machine Learning*, pages 3284–3293. PMLR, 2021.

[7] Kazuyuki Hara, Daisuke Saitoh, and Hayaru Shouno. Analysis of dropout learning regarded as ensemble learning. In *Artificial Neural Networks and Machine Learning–ICANN 2016: 25th International Conference on Artificial Neural Networks, Barcelona, Spain, September 6-9, 2016, Proceedings, Part II 25*, pages 72–79. Springer, 2016. 2

[8] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012. 5

[9] Duy Le Hoai, Eunchae Lim, Eunbin Choi, Sieun Kim, Sudarshan Pant, Guee-Sang Lee, Soo-Huyng Kim, and Hyung-Jeong Yang. An attention-based method for action unit detection at the 3rd abaw competition. *arXiv preprint arXiv:2203.12428*, 2022. 1

[10] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013. 5

[11] Jan Kukačka, Vladimir Golkov, and Daniel Cremers. Regularization for deep learning: A taxonomy. *arXiv preprint arXiv:1710.10686*, 2017. 2

[12] Noam Levi, Itay Bloch, Marat Freytsis, and Tomer Volansky. Noise injection as a probe of deep learning dynamics. *arXiv preprint arXiv:2210.13599*, 2022. 2

[13] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 ieee computer society conference on computer vi-*

*sion and pattern recognition-workshops*, pages 94–101. IEEE, 2010. 1

[14] Ali Mollahosseini, Behzad Hasani, and Mohammad H Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 10(1):18–31, 2017. 4

[15] Antonio Orvieto, Hans Kersting, Frank Proske, Francis Bach, and Aurelien Lucchi. Anticorrelated noise injection for improved generalization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 17094–17116. PMLR, 17–23 Jul 2022. 1, 2

[16] Maja Pantic, Michel Valstar, Ron Rademaker, and Ludo Maat. Web-based database for facial expression analysis. In *2005 IEEE international conference on multimedia and Expo*, pages 5–pp. IEEE, 2005. 1

[17] Sungheon Park and Nojun Kwak. Analysis on the dropout effect in convolutional neural networks. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13*, pages 189–204. Springer, 2017. 5

[18] Mayank Kumar Rusia and Dushyant Kumar Singh. An efficient cnn approach for facial expression recognition with some measures of overfitting. *International journal of information technology*, 13:2419–2430, 2021. 1

[19] Zhiwen Shao, Zhilei Liu, Jianfei Cai, Yunsheng Wu, and Lizhuang Ma. Facial action unit detection using attention and relation learning. *IEEE transactions on affective computing*, 13(3):1274–1289, 2019. 1

[20] Yi Xu, Yuanhong Xu, Qi Qian, Hao Li, and Rong Jin. Towards understanding label smoothing. *arXiv preprint arXiv:2006.11653*, 2020. 1

[21] Xue Ying. An overview of overfitting and its solutions. In *Journal of physics: Conference series*, volume 1168, page 022022. IOP Publishing, 2019.

[22] Haotian Zhang, Lin Zhang, and Yuan Jiang. Overfitting and underfitting analysis for deep learning based end-to-end communication systems. In *2019 11th international conference on wireless communications and signal processing (WCSP)*, pages 1–6. IEEE, 2019. 2

[23] Guoying Zhao, Xiaohua Huang, Matti Taini, Stan Z Li, and Matti PietikäInen. Facial expression recognition from near-infrared videos. *Image and vision computing*, 29(9):607–619, 2011. 1