International Journal of
Advanced Science Computing and Engineering

Journal Homepage www.ijasce.org/index.php/ijasce

Convolutional Neural Network-Based Recognition of Children's Facial Expressions in Response to Gaming

Hadi Santoso*, Genoveva Ferreira Soares, Cristopher Marco Angelo

Informatics Study Program, Faculty of Computer Science, Universitas Mercu Buana Jl. Meruya Selatan No.1, Kembangan, West Jakarta, Jakarta 11610, Indonesia

Corresponding author: hadi.santoso@mercubuana.ac.id

Abstract — **This study explores the use of Convolutional Neural Network (CNN) algorithms for the purpose of recognizing children's facial expressions during gaming activities, with a focus on understanding the emotional consequences of gaming. This study intends to build a robust model and assess the accuracy of CNN in detecting six basic emotions among children aged between 6 and 13 years using our dataset that we collected from children in Timor Leste as many as 600 images and the Children's Real-World Facial Expressions (CFEW) dataset of more than 11,000 images for training data. Then we also use our video data and the LIRIS-CSE dataset from the internet as test data as many as 180 videos and images. The data we obtained were images of children when not playing games and playing games consisting of facial expressions, especially those showing anger, happiness, sadness, fear, surprise, and neutral. This methodology consists of various processes, including data collection, preprocessing, augmentation, model training, and evaluation, with the main goal of identifying patterns and trends in children's emotional responses to games. The results of this study indicate that the final accuracy of detecting children's faces when playing games is 96.78% and the validation data accuracy value is 95.32%. It is proven that the CNN architecture or model used in this research dataset is optimal.**

***Keywords*: children; facial expression recognition; gaming; convolutional neural network (CNN); emotional analysis**

Manuscript received 9 Jul. 2024; revised 17 Jul. 2024; accepted 25 Nov. 2024; Date of publication 31 Dec. 2024.

I. INTRODUCTION

Detecting children's facial expressions is crucial for assessing their gaming activities and grading, as it can significantly impact their development and thought process[1][2][3][4]. Parents should create positive emotions and learn physiological changes through reading their children's faces to assess their understanding. Emotions are associated with cognitive appraisal and motivation, making it important to create an emotionally engaging environment for children to learn[5]. To accomplish all of this, machine learning is required, which can analyze the job in greater depth. As a result, deep learning, a new learning method capable of studying a task in more depth, is now being developed[6].

Deep learning is a subset of machine learning that enables computers to learn based on previous experiences and comprehend commands based on provided concepts [7].

What distinguishes deep learning from machine learning is its deeper learning framework based on the representation of the data to be studied. Deep learning is presently becoming a hot issue in research on object recognition, object detection, speech recognition, and many types of pattern recognition, among other things [8]

Deep learning has altered the pattern recognition research paradigm, which has previously separated feature extraction and classification methods. Deep learning can use both in one structure [9]. Wrong Deep neural networks, also known as Convolutional Neural Networks (CNN), are one technique that employs deep learning approaches. The Convolutional Neural Network (CNN) algorithm's core concept is to mimic the structure of the neural network algorithm, which performs the learning process over numerous layers[10]. In addition, there is a neural algorithm in the background. This network represents a desire to mimic human thinking abilities. As a result, the ability to investigate data in greater depth is essential for applying this neural network approach [11].

Even so, CNN algorithm has a different structure compared to neural network algorithm. To be more specific, neural networks are used in a small portion of the CNN algorithm structure, specifically the classification segment. Meanwhile, in order to learn facial features, this technique, as the name suggests, employs convolution theory for feature processing. Deep learning has demonstrated extraordinary performance in various investigations. This is partly due to its ability to analyze massive datasets and build deeper networks. Aside from that, with the availability of more powerful computing systems, deep learning is becoming more commonly used [12]. CNN is said to be the best model for solving object and facial recognition problems. However, in CNN, it functions similarly to a deep learning model[13][14]. Others have a flaw in which the model computing procedure is rather lengthy. Nevertheless, with the rapid increase in hardware development, it is possible to overcome this issue using Graphical Processing Unit (GPU) technology and a PC with high specifications. Many methods have been proposed to perform this face detection process. Convolutional Neural Network (CNN) is used in reference to do face detection[15].

Another study uses a Region-based Convolutional Neural Network (R-CNN) to detect faces and pedestrians in a video[16]. R-CNN is a CNN invention that is used to tackle a variety of issues, including object detection[17]. The research findings reveal that the proposed method's algorithm is capable of detecting several faces and pedestrians. However, the study has a flaw: it cannot distinguish between real faces (non-spoof) and photos or videos of faces (spoof), which is referred to as facial spoofing [18]. Face spoofing is the process of impersonating someone's face in order to gain unauthorized access to a biometric system. This can be accomplished by presenting a video or image of someone's face on the monitor screen[19]. To address these shortcomings, this research implements the usage of CNN algorithm as an effective technique. Due to its ability to autonomously learn hierarchical representations, Convolutional Neural Networks (CNN) are especially useful for applications involving pictures and spatial data [20]. As a result, they are ideal for tasks that need spatial relationships, such as picture categorization, object detection, and even natural language processing.

Contributions to this study include: (1) a database of facial expressions of children aged 6 to 13 years, which we collected ourselves and used two other datasets, namely the Children's Real-World Facial Expressions (CFEW) dataset and the LIRIS-CSE dataset (2) Using the Grid Search approach, hyperparameter tuning determines the optimal combination of hyperparameters.

II. Materials and Method

A. Basic Concept

This research uses a dataset of 600 images from both primary data or our dataset and LIRIS-CSE dataset. It is split into a training set, augmented with images showing facial expressions related to playing games, and a testing set with 180 images, including some from LIRIS-CSE dataset. This balanced and diverse dataset helps train and evaluate a robust Convolutional Neural Network (CNN) model. The goal is to accurately classify children's facial expressions related to gaming activities using CNN.

B. Flow Diagram

Figure 2.1 explains Flowchart for Child Facial Expression Classification using CNN. The loading model section describes the process. The "loading model" section refers to

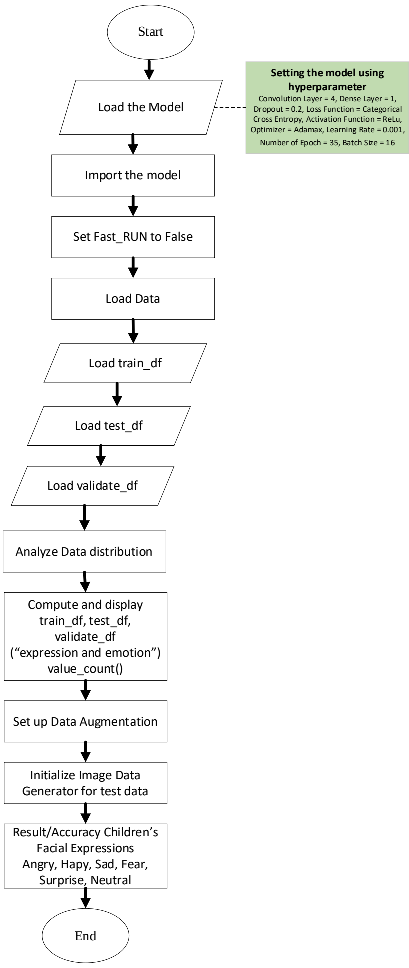


Fig 2.1 Flowchart of Classification System in CNN

the process of importing or initializing the trained model into memory so that it can be used for inference or prediction. In this study, the model is stored in files after training on a large dataset. These files contain the learned model parameters, such as weights and biases, which are important for making predictions. The load_model() function in this case will read the saved model file, load the architecture, and learned weights into memory. The next step is to import the library. By importing these libraries, such as TensorFlow, Keras, and PyTorch and leveraging pre-built functions for model construction, training, and evaluation, it is easy to develop an effective and accurate facial expression recognition system. The next process imports the model, the model has been pre-trained on a general facial expression recognition dataset, the model can be adjusted for children's facial expressions. For a CNN-like model to effectively recognize children's facial expressions, one of the first key steps is data loading. This involves collecting, preprocessing, and feeding image data into the neural network. The importance of this step can be broken down into several key aspects, including: A highly diverse dataset of children's facial expression images representing a wide range of emotional expressions based on age between 6 - 13 years of children in Timor Leste, the Children's Real-World Facial Expressions (CFEW) dataset and the children's drawing dataset in the LIRIS-CSE dataset were collected. A well-curated dataset ensures that the model does not learn biased or limited patterns. Then the facial images need to be standardized before being fed into CNN. This can include resizing, normalization (scaling of pixel values), and augmentation techniques (e.g., rotation, flipping, or brightness adjustment). Proper data preprocessing ensures that the model generalizes better and is insensitive to small inconsistencies in the image. input. Next, Each image or video frame representing a child's face usually needs to be labeled with the appropriate emotion (e.g., happy, sad, angry). These labels are important because they serve as the ground truth during the training process. Correctly labeled data allows CNN learns the relationship between certain facial features (such as eyebrow position, mouth, etc.) and emotional states. The final process is Efficient Data Loading Techniques. Loading image data efficiently is important for training deep learning models. High-resolution images or large datasets can consume a lot of memory. and time. Techniques such as data batching (loading images in small groups) and on-the-fly augmentation (performing transformations such as rotations or flips during the data loading process) help simplify this. Additionally, leveraging frameworks such as TensorFlow or PyTorch can improve data path efficiency through parallel loading and preprocessing. The process of training a CNN to recognize children's facial expressions involves careful preparation, including data collection, preprocessing, model selection, and optimization. Given the complexity and variability of children's emotional expressions, the research's training loading process—which includes selecting appropriate datasets, using effective data augmentation, applying appropriate loss functions and optimization techniques, and rigorous performance evaluation—forms the backbone of the recognition system. effective facial expressions. Proper training allows the model to generalize better, perform more accurately, and ultimately be applied to improve interactions and applications involving children. Load testing in this study refers to the process of evaluating the performance and scalability of a system under varying levels of stress. In the case of CNN applied to child facial expression recognition, load testing is concerned with how well the system can handle large datasets of children's faces, process multiple inputs simultaneously, and maintain accuracy and efficiency as demand increases. Next is the load validation process. a well-validated data set ensures that the accuracy of the model will reflect real-world performance. For children's facial expressions, this is especially important because misclassification of emotions (e.g., confusing sadness with surprise) can lead to unintended consequences in applications such as emotional health monitoring or interactive systems for children. The next process of analyzing the data distribution in a dataset of children's facial expressions is crucial when using Convolutional Neural Networks (CNN) for recognition tasks. The unique features of children's facial expressions, combined with potential biases in the dataset composition, require careful attention. With the data distribution in place, it becomes possible to build more accurate, fair, and generalizable models, improving their performance and real-world applicability across contexts. This is especially important in domains such as emotion detection, interactive learning, and behavioral research, where accurate recognition of children's emotional states is critical to providing appropriate responses and support. The next steps compute and display the process of training, testing, and validating a CNN for recognizing children's facial expressions are critical for ensuring that the model performs well, generalizes across diverse conditions, and is ethically sound. The stages of training allow the model to learn the intricate features of children's faces, testing ensures it can apply this knowledge to new, unseen data, and validation helps fine-tune the model to achieve the best possible performance. The next step is to set up data augmentation. In emotion recognition tasks, collecting a large, diverse dataset of children's facial expressions can be difficult. Data augmentation artificially expands the dataset by applying transformations, making it possible to train CNNs effectively with limited original data. The next step is the initialization of the data generator. Initializing the image data generator for the test data helps standardize the image input, ensuring that the CNN can process the test images correctly and evaluate its performance. In In the case of the "Children's Facial Expressions" dataset in this study, the generator ensures that images of children displaying various facial emotions are properly preprocessed, which is crucial for accurate emotion classification. The final step of this stage is evaluation. The detection of children's facial expressions using CNNs in this study requires evaluation. With deep learning and the availability of larger, more diverse datasets, CNNs provide high accuracy in classifying children's facial expressions in playing games.

C. Data Collection

The collected dataset consists of 600 images of children's facial expressions labeled on each image. There are 600 images and videos of children's faces collected while they are playing games. The Children's Real-World Facial Expressions (CFEW) dataset was then used to generate up to 11,000 images. Figure 2.2 depicts the LIRIS-CSE dataset, which includes 208 videos from 12 children from different cultural backgrounds, while our dataset depicts the contextual context, lighting fluctuations, and natural expressions displayed by children without playing games shown in Figure 2.3. Figure 2.4 shows the natural expressions of our dataset as children play games.

Figure 2.2 Emotional images of children from LIRIS-CSE dataset

Figure 2.3 Emotional images of children from Our dataset

Figure 2.4 Emotional images of children when playing games from Our dataset

D. Data Preprocessing

The collected dataset will undergo preprocessing to remove any noise or unwanted information from the images. The images will also be rescaled from 1600 x 1066 pixels to 600 x 600 pixels to ensure uniformity across the dataset.

E. Data Augmentation

The convolutional neural network model will be trained using synthetic images generated by data augmentation techniques. Rotating, flipping, and cropping will be used on the original photographs to create new images.

F. Model Training

The pre-processed and supplemented dataset will be used to train the convolutional neural network. The model architecture will be developed using past literature and empirical experimentation. To improve model performance, hyperparameters such as learning rate, batch size, and optimizer will be tweaked.

G. Model Evaluation

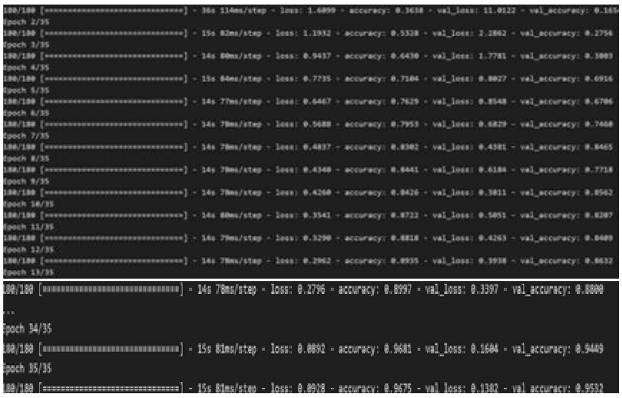
A validation dataset will be used to test the trained model's accuracy, precision, recall, and F1 score. Confusion matrix will also be used to assess the model's performance in terms of false positives and negatives.

III. Results and Discussion

This stage depicts the training progress of a neural network model over numerous epochs, including crucial metrics like loss and accuracy for both the training and validation datasets. It is critical to track how the model's performance changes over time and identify problems such as overfitting. Adjustments to maximize training can be made by analysing changes in metrics from epoch to epoch, such as adjusting hyperparameters or using strategies like early stopping. Overall, monitoring the training progress guarantees that the model is learning properly and progressing toward peak performance

In addition to the evaluation metrics presented above, we also showcase the visual results of our image classification model. During the training of the model, an accuracy of about 96.78% was achieved. This high accuracy is depicted by the blue line in the validation accuracy graph. The validation process, which tests the model's performance on unseen data, showed an accuracy of 95.32%. This training utilized a dataset of 35x35 pixel images and was run over several epochs to refine the model's performance. The next step is data augmentation. In emotion recognition tasks, collecting a large, diverse dataset of children's facial expressions can be difficult. Data augmentation artificially expands the dataset by applying transformations, making it possible to train CNNs effectively with limited original data.

Along with the changes and additions of the epoch value, it is directly proportional to the final accuracy value produced. Where for the epoch value of 35 it produces a final accuracy value of 0.9678 or 96.78% and the final accuracy value of the validation data is 0.9532 or 95.32%. In detail, the training process through an epoch value of 35 can be seen in Figure 3.1:



```
100/100 [==============================] - 36s 114ms/step - loss: 1.6099 - accuracy: 0.3638 - val_loss: 11.0122 - val_accuracy: 0.165
Epoch 2/35
100/100 [==============================] - 15s 82ms/step - loss: 1.1932 - accuracy: 0.5328 - val_loss: 2.2862 - val_accuracy: 0.2756
Epoch 3/35
100/100 [==============================] - 14s 80ms/step - loss: 0.9437 - accuracy: 0.6430 - val_loss: 1.7781 - val_accuracy: 0.3803
Epoch 4/35
100/100 [==============================] - 15s 84ms/step - loss: 0.7735 - accuracy: 0.7104 - val_loss: 0.8027 - val_accuracy: 0.6916
Epoch 5/35
100/100 [==============================] - 14s 77ms/step - loss: 0.6467 - accuracy: 0.7629 - val_loss: 0.8548 - val_accuracy: 0.6706
Epoch 6/35
100/100 [==============================] - 14s 78ms/step - loss: 0.5688 - accuracy: 0.7953 - val_loss: 0.6829 - val_accuracy: 0.7460
Epoch 7/35
100/100 [==============================] - 14s 78ms/step - loss: 0.4837 - accuracy: 0.8302 - val_loss: 0.4381 - val_accuracy: 0.8465
Epoch 8/35
100/100 [==============================] - 14s 78ms/step - loss: 0.4340 - accuracy: 0.8441 - val_loss: 0.6184 - val_accuracy: 0.7718
Epoch 9/35
100/100 [==============================] - 14s 78ms/step - loss: 0.4260 - accuracy: 0.8426 - val_loss: 0.3811 - val_accuracy: 0.8562
Epoch 10/35
100/100 [==============================] - 14s 80ms/step - loss: 0.3541 - accuracy: 0.8722 - val_loss: 0.5051 - val_accuracy: 0.8207
Epoch 11/35
100/100 [==============================] - 14s 79ms/step - loss: 0.3290 - accuracy: 0.8818 - val_loss: 0.4263 - val_accuracy: 0.8409
Epoch 12/35
100/100 [==============================] - 14s 78ms/step - loss: 0.2962 - accuracy: 0.8935 - val_loss: 0.3938 - val_accuracy: 0.8632
Epoch 13/35
100/100 [==============================] - 14s 78ms/step - loss: 0.2796 - accuracy: 0.8997 - val_loss: 0.3397 - val_accuracy: 0.8880
...
Epoch 34/35
100/100 [==============================] - 15s 81ms/step - loss: 0.0892 - accuracy: 0.9681 - val_loss: 0.1604 - val_accuracy: 0.9449
Epoch 35/35
100/100 [==============================] - 15s 81ms/step - loss: 0.0928 - accuracy: 0.9675 - val_loss: 0.1382 - val_accuracy: 0.9532
```

Fig 3.1 Source code of Loss & accuracy

Figure 3.2 shows that the accuracy and validation values are directly proportional according to the increase in the epoch value. The final accuracy results obtained are 96.78% and the validation data accuracy value is 95.32%.

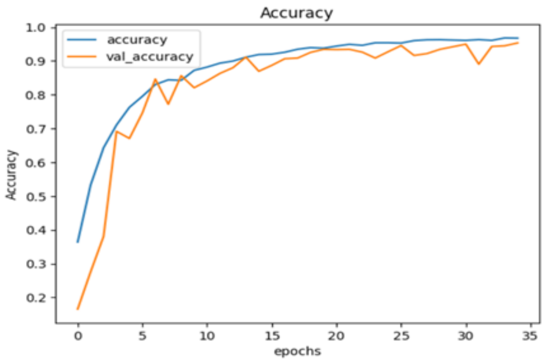


Fig 3.2 Model Accuracy

Figure 3.3 shows a visualization of the accuracy and loss graphs at each epoch. On the graph, we have two lines, one for training loss (blue) and one for validation loss (yellow). In the experiment using the CNN architecture with 35 epochs, the loss value on the graph decreased drastically in both training loss and validation loss. The decrease in the loss value for training started at the 2nd epoch, while the loss for validation started at the 12th epoch, after which the loss values for training and validation stabilized at numbers below 0.3.

Table 3.1 of Loss and Accuracy Testing Results

Attempt	***Loss***	***Accuracy***
1	0.27	0.8997
2	0.032	0.9475
3	0.028	0.9678

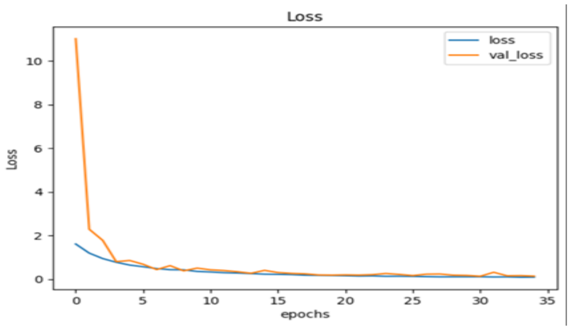


Fig 3.3 Model Loss

The Testing results in Table 3.1 for loss and accuracy in the three experiments experienced differences, in the first experiment the loss testing value was at 0.27 or 27%, and for the accuracy testing value of the first experiment it was 89.97%. In the second experiment the loss testing value was at 0.032 or 3.2%, and for the accuracy testing value of the second experiment it was 94.75%. In the third experiment the loss testing value was at 0.028 or 2.8%, and for the accuracy testing value of the third experiment it obtained 96.78%.

This means that the architecture or model used for this research dataset is more optimal, because when testing the model with a dataset that is not in training and validation, the model is more optimal and accurate and precise in reading or detecting datasets in the form of images, to classify into classes of Children's facial expressions against the influence of playing games.

Figure 3.4 shows the final result of emotion detection on the CNN algorithm

Fig 3.4 Emotion Detection Results

IV. Conclusion

In conclusion, employing CNN algorithms to identify children's facial expressions while gaming is a significant step forward in understanding how games affect children emotionally. We may learn about their moods and level of engagement in the game by examining their facial expressions. This allows game developers to create better games that kids would like more. The studies of this subject also provide insights into how gaming affects children's learning and social abilities. We can tell which elements of a game they find tough or entertaining by observing their facial expressions. This data can be used to improve the effectiveness and entertainment value of instructional games. Overall, using CNN algorithms to analyze children's facial expressions while gaming is an effective technique to learn more about how games affect children emotionally and socially. It may result in better games and support for children in need.

Acknowledgment

The authors would like to express their gratitude to the Faculty of Computer Science, Informatics Study Program, Universitas Mercu Buana, for their support and resources provided throughout this research.

References

[1] D. Piontkowski and R. Calfee, "Child Emotion Detection Through Facial Expression Recognition Using Machine Learning," *Attention and Cognitive Development*, pp. 297–329, 2023, doi: 10.1007/978-1-4613-2985-5_11.

[2] U. Laraib, A. Shaukat, R. A. Khan, Z. Mustansar, M. U. Akram, and U. Asgher, "Recognition of Children's Facial Expressions Using Deep Learned Features," *Electronics (Switzerland)*, vol. 12, no. 11, Jun. 2023, doi: 10.3390/electronics12112416.

[3] E. Guo, "A Review of the Facial Emotion Recognition in Children with Autism," in *Proceedings of the 2022 5th International Conference on Humanities Education and Social Sciences (ICHESS 2022)*, Atlantis Press SARL, 2022, pp. 2411–2418. doi: 10.2991/978-2-494069-89-3_277.

[4] J. G. Negrão *et al.*, "The Child Emotion Facial Expression Set: A Database for Emotion Recognition in Children," *Front Psychol*, vol. 12, Apr. 2021, doi: 10.3389/fpsyg.2021.666245.

[5] Y. Wang, Q. Luo, Y. Zhang, and K. Zhao, "Synchrony or asynchrony: development of facial expression recognition from childhood to adolescence based on large-scale evidence," *Front Psychol*, vol. 15, 2024, doi: 10.3389/fpsyg.2024.1379652.

[6] Y. Ihza and D. Lelono, "Face Expression Classification in Children Using CNN," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 16, no. 2, p. 159, Apr. 2022, doi: 10.22146/ijccs.72493.

[7] M. Soori, B. Arezoo, and R. Dastres, "Artificial intelligence, machine learning and deep learning in advanced robotics, a review," Jan. 01, 2023, *KeAi Communications Co.* doi: 10.1016/j.cogr.2023.04.001.

[8] M. M. Taye, "Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions," May 01, 2023, *MDPI*. doi: 10.3390/computers12050091.

[9] K. Reghunandanan, V. S. Lakshmi, R. Raj, K. Viswanath, C. Davis, and R. Chandramohanadas, "A Convolutional Neural Network- Based Deep Learning To Detect Reticulocytes From Human Peripheral Blood," *Intell Based Med*, vol. 10, Jan. 2024, doi: 10.1016/j.ibmed.2024.100175.

[10] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artif Intell Rev*, vol. 57, no. 4, Apr. 2024, doi: 10.1007/s10462-024-10721-6.

[11] J. Serey *et al.*, "Pattern Recognition and Deep Learning Technologies, Enablers of Industry 4.0, and Their Role in Engineering Research," Feb. 01, 2023, *MDPI*. doi: 10.3390/sym15020535.

[12] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artif Intell Rev*, vol. 57, no. 4, Apr. 2024, doi: 10.1007/s10462-024-10721-6.

[13] M. Ali Sultan *et al.*, "Brain Tumor Detection and Classification Using Fine-Tuned CNN with ResNet50 and EfficientNet," *International Journal of Informatics and Computation (IJICOM)*, vol. 6, no. 1, 2024, doi: 10.35842/ijicom.

[14] H. Santoso, I. Hanif, and H. Magdalena, "A Hybrid Model for Dry Waste Classification using Transfer Learning and Dimensionality Reduction," vol. 8, no. 2, 2024, doi: 10.62527/joiv.8.2.1943.

[15] E. S. Agung, A. P. Rifai, and T. Wijayanto, "Image-based facial emotion recognition using convolutional neural network on emognition dataset," *Sci Rep*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-65276-x.

[16] Y. Permanasari, B. N. Ruchjana, S. Hadi, and J. Rejito, "Innovative Region Convolutional Neural Network Algorithm for Object Identification," Dec. 01, 2022, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/joitmc8040182.

[17] R. Zimmer, M. Sobral, and H. Azevedo, “Hybrid Models for Facial Emotion Recognition in Children,” Aug. 2023, [Online]. Available: http://arxiv.org/abs/2308.12547

[18] Y.-J. Zhang, J.-Y. Chen, and Z.-M. Lu, “Face anti-spoofing detection based on color texture structure analysis,” *Taiwan Ubiquitous Information*, vol. 7, no. 2, 2022.

[19] P. Jaswanth, P. Y. chowdary, and M. V. S. Ramprasad, “Deep learning based intelligent system for robust face spoofing detection using texture feature measurement,” *Measurement: Sensors*, vol. 29, Oct. 2023, doi: 10.1016/j.measen.2023.100868.

[20] C.-Y. Weng, “Land-Use Classification via Transfer Learning with a Deep Convolutional Neural Network,” *Journal of Intelligent Learning Systems and Applications*, vol. 14, no. 02, pp. 15–23, 2022, doi: 10.4236/jilsa.2022.142002.