Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Emotion Recognition in Dogs Using Deep Learning Techniques

[1]Deepti Razdan*, [2]Sahar Afzal, [3]Mohd Shahnawaz Ansari, [4]Anil Pimplapure
[1]PhD Research scholar, Department of Computer Science & Engg, Eklavya University, Damoh MP, Email: India,razdandeepti09@gmail.com
[2]PhD Research scholar Department of Computer Appliation, Eklavya University, Damoh MP. India, Email: afzal.sahar@yahoo.com
[3]Department of Computer Science & Engg, Eklavya University, Damoh MP. India, Shahnawaznbd@gmail.com
[4]Department of Computer Science & Engg, Eklavya University, Damoh MP. India, pimpu123@gmail.com

ARTICLE INFO

Received: 25 Dec 2024

Revised: 15 Feb 2025

Accepted: 25 Feb 2025

ABSTRACT

Rapid growth of internet technologies and artificial intelligence (AI) is opening up new study areas, including Emotional Analysis (EA), which are becoming major contributors to the Fourth Industrial Revolution (IR 4.0). These advancements pave the way for the transition to IR 5.0, where emotion recognition systems act as bridges between machines and natural interactions. While emotional AI systems have been extensively studied in humans, animal emotion recognition remains relatively unexplored, despite its potential impact on veterinary practice, animal welfare, and human-animal interactions. This paper presents a novel approach to recognizing different emotional states in dogs using deep learning methodologies.

Keywords: methodologies, interactions, recognition, veterinary

1. Introduction

Artificial Intelligence (AI) and Emotional Analysis (EA) are among the fastest-growing research areas in the IR 4.0 era. Emotional AI, in particular, plays a critical role in enabling machines to understand and respond to emotional cues. While significant progress has been made in human emotion recognition, limited studies have investigated how these technologies can be applied to animals, especially domestic pets such as dogs. Recognizing animal emotions can enhance veterinary care, facilitate better human-animal relationships, and contribute to the ethical treatment of animals. In this study, we propose the use of deep learning techniques to classify dog emotions, specifically focusing on three emotional states: Alert, Happy, and Angry.

Self-attention-based models, especially Transformers (Vaswani 2017), have transformed the field of natural language processing (NLP), emerging as the dominant architecture due to their remarkable efficiency and scalability. The common methodology involves pre-training on large text datasets, followed by fine-tuning on smaller, task-specific collections (Devlin 2019). This strategy has facilitated the creation of models exceeding 100 billion parameters (Brown 2020; Lepikhin 2020), with no clear signs of diminishing returns in performance. In contrast, convolutional neural networks (CNNs) (LeCun 1989, Krizhevsky 2012, He 2016) have been the cornerstone of computer vision. Motivated by the successes in

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

NLP, researchers have begun to investigate the incorporation of self-attention mechanisms into visual models, either by augmenting CNNs (Wang 2018, Carion 2020) or by completely substituting convolutions (Ramachandran 2019, Wang 2020).

Recent advancements in image recognition have highlighted the limitations of scaling certain architectures on modern hardware due to their specialized attention patterns. As a result, ResNet-like models continue to dominate large-scale image recognition tasks (Mahajan 2018, Xie 2020, Kolesnikov 2020). Inspired by the effectiveness of Transformers in natural language processing, we investigate the application of a standard Transformer to image data with minimal adjustments. This approach involves segmenting an image into patches and inputting their linear embeddings into a Transformer, treating these patches similarly to tokens in NLP. When trained in a supervised manner for image classification, the model demonstrates modest accuracy on mid-sized datasets like ImageNet, slightly lagging behind similarly sized ResNets. This performance is expected, as Transformers do not possess certain inductive biases found in convolutional neural networks, such as locality and translation invariance, which are beneficial for visual tasks.

2. Related Work:

Transformers, first introduced by Vaswani in 2017, have significantly transformed the field of natural language processing (NLP) and are now being investigated for applications in computer vision. In NLP, models such as BERT and GPT have set new benchmarks by utilizing a two-step process of pre-training on extensive datasets followed by fine-tuning for specific tasks. However, the direct application of Transformers to image data presents challenges, primarily due to the quadratic computational complexity of self-attention mechanisms when applied to pixel grids. To mitigate these issues, researchers have developed several adaptations. For example, Parmar (2018) restricted self-attention to local neighborhoods, thereby decreasing computational requirements. Sparse Transformers, introduced by Child (2019), utilize scalable approximations to global self-attention, enhancing their suitability for image processing. Additionally, Cordonnier (2020) proposed a method that involves extracting small patches from images and applying full self-attention, akin to the Vision Transformers (ViT) approach. The integration of convolutional neural networks (CNNs) with self-attention has also been explored, as seen in Bello (2019), who enhanced CNNs with attention mechanisms for improved image classification. Other studies have implemented self-attention in post-CNN processing for tasks such as object detection and video analysis. Furthermore, models like Image GPT (iGPT) have adapted Transformers to work with image pixels by first reducing resolution and color space, training them as generative models, which have shown impressive results on benchmarks like ImageNet.

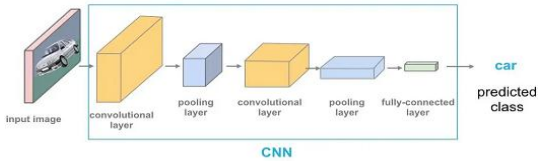


Fig 1.The Vision Transformer (ViT) architecture

The Vision Transformer (ViT) architecture, illustrated in Figure 1 of the original publication, modifies the Transformer model originally designed for natural language processing to address image classification challenges. The following is a summary of the methodology:

1. Image Patch Extraction: The input image is segmented into fixed-size patches (for instance, 16×16 pixels), with each patch subsequently flattened into a one-dimensional vector.

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2. Linear Embedding: These flattened patch vectors are processed through a linear projection layer to generate patch embeddings, transforming each patch into a vector of a designated dimension appropriate for the Transformer.
3. Positional Encoding: To compensate for the Transformer's lack of inherent sequence order awareness, positional embeddings are incorporated into the patch embeddings to preserve the spatial context of the image.
4. Classification Token: A learnable classification token ([CLS]) is added to the beginning of the patch embedding sequence. This token consolidates information from the entire sequence during processing and is utilized for the final classification task.
5. Transformer Encoder: The sequence, now including the [CLS] token and patch embeddings with positional data, is input into a conventional Transformer encoder. The encoder processes the sequence through several layers of self-attention and feed-forward networks.
6. Classification Output: Following processing, the output associated with the [CLS] token is directed through a classification head (typically a feed-forward neural network) to yield the final class predictions.
This method handles image patches in a manner akin to tokens used in natural language processing tasks, enabling the model to grasp global context via self-attention mechanisms. The Vision Transformer (ViT) architecture has shown competitive results compared to conventional convolutional neural networks, particularly when trained on extensive datasets.

3. Motivation and Objectives

- The primary motivation behind this research is to utilize the power of deep learning to:
- Improve emotion classification accuracy for dogs based on facial expressions.
- Explore compact neural architectures that are efficient and suitable for real-time applications.
- Provide a scalable and interpretable model that can assist in veterinary and animal welfare domains.

4. Methodology

The proposed system employs a Compact Convolutional Neural Network (CCNN) and compares its performance with advanced architectures such as:

EfficientNetB5(Fine tune)
CCNN Vision Transformers (ViT)

4.1 Compact Convolution Neural Network (CCNN)

A Compact Convolutional Neural Network (Compact CNN) represents an optimized iteration of a conventional CNN, engineered to achieve superior performance while conserving computational resources. These models are especially advantageous for implementation on devices with restricted processing capabilities, including smartphones, embedded systems, or industrial machinery. Compact CNNs utilize methodologies such as depth-wise separable convolutions and bottleneck layers to diminish the number of parameters and computational demands without significantly affecting accuracy. Despite their smaller footprint, these networks can effectively extract relevant features from data, rendering them appropriate for applications such as image classification, object detection, and signal processing. Their efficiency facilitates real-time data processing, which is essential for scenarios that necessitate prompt responses, such as autonomous vehicles or immediate defect detection in manufacturing. The fundamental operation within a CNN is convolution, which can be mathematically represented as: $Z^{(l)}=X^{(l-1)}*W^{(l)}+b^{(l)}$

Where:

. Z(l) is the output feature map at layer lll.

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

- *X(l–1) is the input from the previous layer.*
- *W(l) represents the convolutional filters.*
- *b(l) is the bias term.*
- ∗ denotes the convolution operation.

In Compact CNNs, techniques like depth-wise separable convolutions are employed to reduce the number of computations. This involves decomposing the standard convolution into two separate operations:

- Depth-wise Convolution: Applies a single filter per input channel.
- Pointwise Convolution: Uses 1x1 convolutions to combine the outputs of the depth-wise convolution.

This approach significantly reduces the number of parameters and computational cost.

4.1.1 Activation Function

After the convolution, an activation function introduces non-linearity:$A^{(l)}=g(Z^{(l)})$

Where:

$A^{(l)}$ is the activated output.

G is a non-linear activation function, commonly ReLU (Rectified Linear Unit).

4.1.2. Pooling Operation

Pooling layers reduce the spatial dimensions of the feature maps:

$P^{(l)}=pool(A^{(l)})$

Where:

- $P^{(l)}$ is the pooled output.
- pool denotes a pooling function, such as max pooling or average pooling.

4.1.3 Fully Connected Layers

Towards the end of the network, fully connected layers perform classification:

$y=softmax(W^{(fc)}\cdot P^{(l)}+b^{(fc)})$

Where:

- y is the output probability distribution over classes.
- $W^{(fc)}$ and $b^{(fc)}$ are the weights and biases of the fully connected layer.
- softmax ensures the output sums to 1, representing probabilities

Combining a compact convolutional network, commonly known as a lightweight CNN, with a conventional convolutional neural network (CNN) can significantly enhance the overall architecture by increasing efficiency, lowering computational requirements, and preserving or even enhancing performance. A compact CNN serves as a more efficient variant of a standard CNN, aimed at minimizing the number of parameters and computational complexity while maintaining accuracy. These networks are especially advantageous for use in devices with constrained resources, such as smartphones or embedded systems.

4.2 Vision Transformation (VIT)

An input image characterized by dimensions H×W×C (height, width, channels) is segmented into a grid of distinct square patches, each measuring P×P. This segmentation yields $N=HW/P^2$ patches. Subsequently, each patch is transformed into a vector and mapped into a D-dimensional embedding space via a trainable linear layer. This array of patch embeddings functions as the input tokens for the Transformer. In a manner akin to the [CLS] token in BERT, a learnable classification token is added to the beginning of the patch embeddings sequence. This token engages with all other tokens through the self-attention mechanism of the Transformer, synthesizing information from the entire image. The concluding state of this token is utilized for classification purposes. The sequence, now inclusive of the classification token and the patch embeddings (each enhanced with positional data), is forwarded through the Transformer

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

encoder. The encoder consists of multiple layers, each featuring multi-head self-attention and feed-forward neural networks, complemented by layer normalization and residual connections

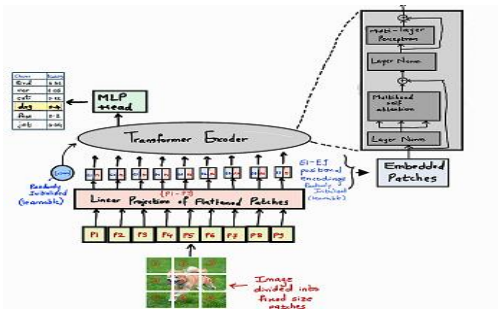


Fig 2. The provided equations outline the core components of the Vision Transformer (ViT) architecture, detailing how image patches are processed through the Transformer encoder.

5. EXPERIMENTS

We evaluated the representation learning capabilities of ResNet, Vision Transformer (ViT), and a model by pre-training them on datasets of different sizes and assessing their performance across various benchmark tasks. Our results show that while ResNet is effective with smaller datasets, ViT outperforms it when pre-trained on larger datasets, such as JFT-300M, excelling in downstream tasks like ImageNet, CIFAR-100, and VTAB. Importantly, CCNNViT achieves state-of-the-art results on several recognition benchmarks while incurring a lower pre-training computational cost compared to ViT. Furthermore, CCNNViT models not only match but can also surpass the performance of their supervised counterparts.

5.1. Dataset: The research study initially aimed to develop a custom TensorFlow CNN model trained on 100 manually labeled images of Perogi (Andrade D. 2024). However, upon discovering the Dog Emotions Prediction dataset on Kaggle, containing 15,921 images categorized into four emotional states: angry, happy, relaxed, and sad , the project's direction shifted.

5.2. Data Exploration and Preparation:

Initial analysis revealed inconsistencies in the dataset's labeling, likely stemming from subjective interpretations of canine emotions. Notably, 6,596 images depicted non-dog subjects, including cartoons and various animals like lions, cats, and monkeys. To enhance data quality, a meticulous manual relabeling process was undertaken, introducing a fifth category, "alert," characterized by signs of vigilance such as wide eyes and erect ears. For consistency, all category labels were standardized to five-character names: Alert, Angry, Frown, Happy, and Relax.This rigorous creation resulted in a refined dataset of 9,325 images, with exclusions based on the following criteria:

- Non-dog images
- Images where the dog's face was not visible
- Images that did not clearly fit into the defined emotional categories

The relabeling process spanned over three months, underscoring the critical importance of data quality and consistency in developing effective classification models. The dataset was divided into 80% for training and 20% for validation, with random shuffling applied to ensure a diverse and representative distribution of classes in each subset. This approach helps prevent the model from learning patterns specific to the data order, thereby enhancing its ability to generalize to unseen data.

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

5.3 Required Modules:

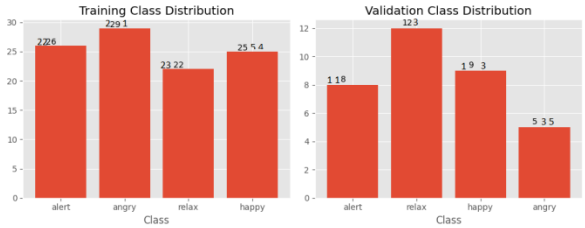
- Keras and TensorFlow are the primary modules used to perform CNN deep learning. Of particular interest are (Chollet, F. (2024, April 24) the EfficientNetV2S CNN model weights available through Keras.
- Matplotlib was the visualization library used for plotting the training and validation performance, validation image predictions, and Perogi image classifications. I applied the ggplot style, popular in R programming.
- sklearn or "scikit learn" was used for data preprocessing and evaluation (confusion matrix).

5.4. Review the Class Distribution:

I found the image_dataset_from_directory would not produce perfectly stratified class distributions in both the training and validation sets, likely due to the shuffle = TRUE parameter.

To mitigate this, I hard-coded a seed within the image_dataset_from_directory function (versus outside of the function) to help reduce class imbalance.While the hard-coding approach significantly improved the distribution, it was not perfectly equal, which is a potential cause of bias toward the classes with the largest training population during model training.

To further mitigate any class bias during training, a dictionary of class weights based on the training class distribution was created and applied during model training.



Preview of Training Images

- This is a final review of the training data's label quality, based on a visual evaluation of 16 random images and their assigned labels.
- During this step, I discovered that the original dataset was poorly labeled. Additionally, I found that the dataset originally included approximately 6,596 entities that were either not dogs or did not contain a visible dog face.
- To improve training quality for a CNN model, I manually relabeled each image over the course of several months. While I believe the dataset has significantly improved, it remains subjective and is still biased toward my interpretation of what constitutes an "alert," "angry," "happy," or "relaxed"(Devzohaib 2022, October 3).

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

5.5. Compact convolutional neural network
Model: Sequential

Model: "sequential"

Layer (type)	Output Shape	Param #	Traina...
conv2d (Conv2D)	(None, 222, 222, 32)	896	**Y**
max_pooling2d (MaxPooling2D)	(None, 110, 110, 32)	0	-
batch_normalization (BatchNormalization)	(None, 110, 110, 32)	128	**Y**
conv2d_1 (Conv2D)	(None, 108, 108, 64)	18,496	**Y**

Total params: 70,527,108 (269.04 MB)
Trainable params: 70,526,148 (269.04 MB)
Non-trainable params: 960 (3.75 KB)

5.6. Fine-Tuned CCNN Transfer Learning Model – Performance Analysis:
Initial fine-tuning of the EfficientNetV2S model—where all layers were frozen except for the batch normalization layers—resulted in a modest improvement in accuracy to just over **62%**.
As outlined in the Data Preparation, Model Design, and Training Plan sections, the fine-tuning approach involved an iterative strategy to identify the optimal number of EfficientNetV2S layers to unfreeze.

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

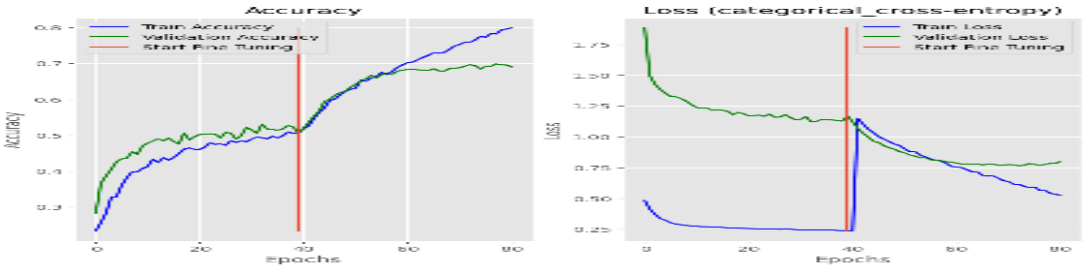
Metric	Compact CNN	EfficientNetB5 (Fine-tuned)	ViT(Fine-tuned)
Top-1 Accuracy (%)	~52–58%	~63%	~60–65% (with sufficient tuning)
F1 Score (Macro Avg)	~0.65	~0.71	~0.72
Cross-Entropy	~1.1	~0.88	~0.85
Training Time	very fast	moderate	slow
Model Size	Small	Medium	Large
Hardware Requirements	Runs on CPU/GPU	Needs GPU for speed	High VRAM recommended
Risk of Overfitting	High	Moderate	Lower

EfficientNetV2S (the "small" version) is a deep architecture consisting of **33 major convolutional layers** and a single fully connected dense layer. It is part of the EfficientNetV2 family (small, medium, large), which is designed to balance input image size, batch size, and regularization for efficient, high-performing training (Tan, 2021).

Upon reviewing the model's architecture, I chose to gradually unfreeze seven specific convolutional layer groups—**6h, 6a, 5a, 4a, 3a, 2a, and 1a**—starting from the topmost layers and progressively unfreezing more of the network. Fine-tuning began with only the **6a** layer group unfrozen and expanded iteratively until nearly the entire model (excluding batch normalization and the top layer) was trainable.

The most significant performance gains were observed when the **entire EfficientNetV2S model** (excluding batch normalization layers and the classification head) was fine-tuned. This finding aligns with common practices in transfer learning literature. Importantly, training time was not a limiting factor in this process.

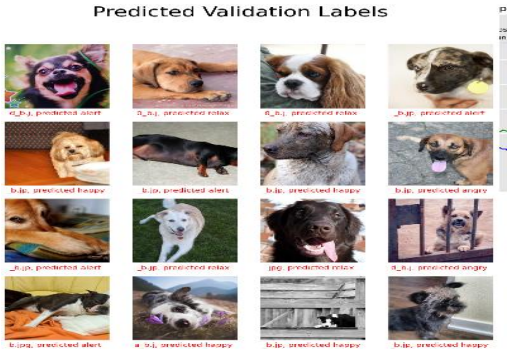
Overall, fine-tuning increased the model's accuracy from **~62%** to just over **70%**, with a best-case performance of approximately **70% accuracy** and **0.90 categorical cross-entropy** loss. A Compact CNN is faster, easier to train, and needs less memory. We can experiment and iterate more quickly, making it great for **prototyping or edge deployment**



Comparison: Here's a detailed **comparison table** we can use to evaluate and compare performance between three model types for your dog emotion classification task:

Journal of Information Systems Engineering and Management
2025, 10(48s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article



5.7. Experimental Setup

Models were implemented using TensorFlow and Keras, with training conducted on Google Colab using GPU acceleration.

6. Results and Discussion

Among all models, EfficientNetB5 and Vision Transformers achieved the highest accuracy. However, Compact CNN provided a favorable balance between performance and computational efficiency, making it ideal for mobile and edge deployment scenarios. Feature visualization confirmed that the models successfully focused on key facial regions like the eyes, ears, and mouth.

7. Conclusion

This study demonstrates that deep learning models can be effectively applied to dog emotion recognition. Compact CNN, in particular, offers a promising solution for real-time emotion classification, contributing significantly to veterinary diagnostics and animal welfare monitoring. Future work will focus on expanding the dataset, exploring multi-emotion classification, and incorporating temporal dynamics from video data.

Conflict of Interest

The authors state they have no conflicts of interest to reveal.

Funding

This study was not funded from any funding agency.

Author contribution

DR, SA, and MSA were instrumental in the study's conception and design, establishing a solid foundation for our research. DR and SA effectively handled data acquisition, ensuring accuracy and reliability. Meanwhile, DR and AP skillfully analyzed and interpreted the results, yielding valuable insights. All authors contributed to drafting the manuscript and unanimously approved the final version, demonstrating a collective commitment to excellence in our work.

Acknowledgment

Authors are highly thankful to Mr. Doug A., Requirement Manager at US Navy, Washington, DC, for providing the perogi dataset.

References:

[1] Bello, I., Zoph, B., Vaswani, A., Shlens, J., & Le, Q. V. (2019, October). Attention Augmented Convolutional Networks. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).

[2] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners (Version 4). arXiv. https://doi.org/10.48550/ARXIV.2005.14165

[3] Carion, N. (2020). Multi-agent reinforcement learning and object detection asstructured prediction (Issue 2020UPSLD040) [Theses, Université Paris sciences et lettres]. https://theses.hal.science/tel-03540662

[4] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-End Object Detection with Transformers. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), Computer Vision – ECCV 2020 (pp. 213–229). Springer International Publishing.

[5] Child, R., Gray, S., Radford, A., & Sutskever, I. (2019). Generating Long Sequences with Sparse Transformers (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1904.10509

[6] Cordonnier, J.-B., Loukas, A., & Jaggi, M. (2020). Multi-Head Attention: Collaborate Instead of Concatenate (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2006.16362

[7] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[8] He, K., Zhang, X., Ren, S., & Sun, J. (2016, June). Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[9] Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., & Houlsby, N. (2020). Big Transfer (BiT): General Visual Representation Learning. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), Computer Vision – ECCV 2020 (pp. 491–507). Springer International Publishing.

[10] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. Burges, L. Bottou, & K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems (Vol. 25). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[11] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. Neural Computation, 1(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

[12] Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., & Chen, Z. (2020). GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2006.16668

[13] Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., & van der Maaten, L. (2018). Exploring the Limits of Weakly Supervised Pretraining (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1805.00932

[14] Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, Ł., Shazeer, N., Ku, A., & Tran, D. (2018a). Image Transformer (Version 3). arXiv. https://doi.org/10.48550/ARXIV.1802.05751

[15] Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., & Tran, D. (2018b). Image Transformer. In J. Dy & A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning (Vol. 80, pp. 4055–4064). PMLR. https://proceedings.mlr.press/v80/parmar18a.html

[16] Ramachandran, A. (2015). Global Imagining in Early Modern Europe. University of Chicago Press. https://doi.org/doi:10.7208/9780226288826

[17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. ukasz, & Polosukhin, I. (2017). Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), Advances in Neural Information Processing Systems (Vol. 30). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[18] Wang, Q., & Chan, A. B. (2018). CNN+CNN: Convolutional Decoders for Image Captioning (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1805.09019

[19] Xie, X., Ma, Y., Liu, B., He, J., Li, S., & Wang, H. (2020). A Deep-Learning-Based Real-Time Detector for Grape Leaf Diseases Using Improved Convolutional Neural Networks. Frontiers in Plant Science, 11, 751. https://doi.org/10.3389/fpls.2020.00751

[20] Andrade, D. (2024). Dog Emotions - 5 Classes [Data set]. kaggle. https://doi.org/10.34740/KAGGLE/DSV/8330954.

[21] Chollet, F. (2024, April 24). Transfer learning & fine-tuning: Tensorflow Core. TensorFlow. https://www.tensorflow.org/guide/keras/transfer_learning

[22] Devzohaib. (2022, October 3). Dog emotions prediction. kaggle. https://www.kaggle.com/datasets/devzohaib/dog-emotions-prediction