# ADOBE | Image Classification and Artifact Detection

## Technical Report

Inter-IIT Tech Meet 13.0

Team_83

## Introduction

Advances in artificial intelligence and machine learning have transformed digital content creation. Diffusion-based generative models like Adobe Firefly [43], Stable Diffusion [44], and Midjourney [45] have enabled the production of highly realistic synthetic media, revolutionizing creative workflows across industries. These tools offer unprecedented opportunities to automate content creation, enhance digital experiences, and inspire innovation.

However, the proliferation of synthetic media presents significant challenges to authenticity and trust. The misuse of these technologies—such as the spread of deepfakes and manipulated visuals—threatens the integrity of digital content across various sectors, including journalism, e-commerce, and personal identification. As the distinction between real and AI-generated media becomes increasingly blurred, the ability to reliably differentiate between the two has become vital.

Some critical areas impacted by the misuse of generative tools include:

- **Journalism:** Deepfakes and falsified visuals undermine the credibility of news, creating an urgent need for robust verification tools.
- **E-Commerce:** Manipulated product images can erode consumer trust, necessitating systems to ensure media authenticity.
- **Privacy:** The misuse of synthetic media poses risks to personal identity and privacy, emphasizing the importance of reliable detection systems.

This project aims to advance detection methods for AI-generated images, with a particular focus on diffusion-based models. The proposed approach integrates four different models to develop a system capable of identifying whether an image is real or AI-generated, while providing accurate reasoning to support the detection.

## Key Challenges

(1) **Quality and quantity of dataset given:** Although the CIFAKE dataset was large (60k real and 60k fake), the low-resolution and blurred nature of the images impacts the model's ability to generalise on new data and make predictions.

(2) **Verification of artifacts in the downsampled images:** AI-generated images have subtle artifacts which become difficult to detect when the images are downsampled. When downscaled, certain pixel-value anomalies and high-frequency artifacts may get smoothened out resulting in false negatives and failure to detect key artifacts.

(3) **Difficulty in converting visual data to text for interpretation:** Convolutional Neural Networks operate as "black boxes", i.e, their internal decision making processes are not easily interpretable. They do achieve high accuracy in classification tasks but explaining 'why' the model classified an image as real or fake is a difficult task.

(4) **Position of artifact in image:** Identifying the artifacts was a difficult task itself, but locating those artifacts in the downsampled image was even more difficult. Since the image was downsampled features were not very visible. To tackle this the image was upsampled and then the explanation was generated which is explained in detail **here**.

(5) **Image Size and Indistinguishability:** The 32×32 image size, combined with the realism of advanced generative models, makes distinguishing real from AI-generated content highly challenging.

(6) **Inference Time and Computational Costs:** Ensuring real-time processing on mobile devices requires lightweight models with minimal computational overhead.

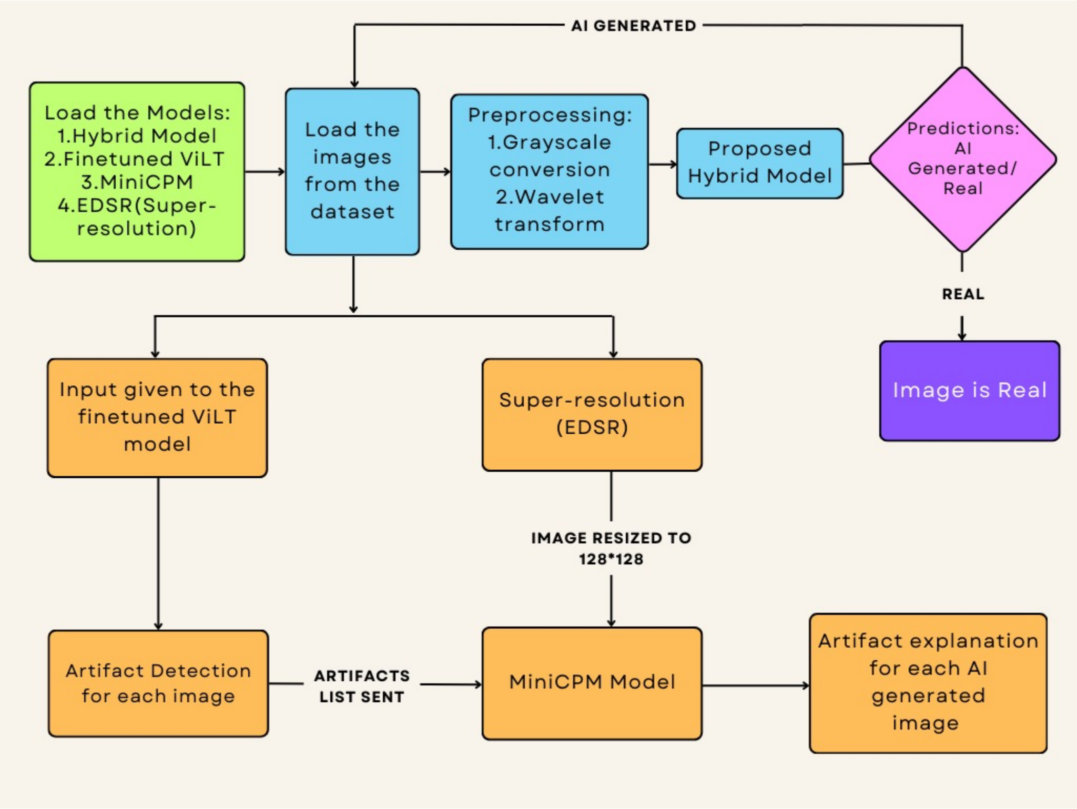


**Figure 1: Proposed Model (EDSR model from [3]).**

(7) **Bias and Explainability:** Dataset biases (e.g., color distributions) and the need for transparent, interpretable decisions pose significant challenges.

# Methodology

Figure 1 outlines the flow of the methodology and the overall process. It begins with loading the required models and dataset. Following this, the proposed hybrid model classifies each input image as either "REAL" or "FAKE." For images identified as "FAKE," the ViLT model fine tuned on VQA (Visual Question Answering) dataset is applied to detect the specific artifacts present. Subsequently, a list of identified artifacts, along with a super-resolved and upsampled version of the image, is passed to the MiniCPM [9] Vision Language Model. This model generates detailed explanations for each artifact detected.

## 1 TASK-1

Detecting AI-generated images is a complex task due to the increasing sophistication of modern generative techniques. To tackle this challenge, we proposed a hybrid model that integrates Convolutional Neural Networks (CNNs) with wavelet-based feature extraction. This combination capitalizes on the ability of CNNs to capture spatial features and the wavelet transform's strength in analyzing frequency-domain information. By leveraging these complementary methodologies, the model performs a comprehensive analysis of image artifacts, enabling robust detection of synthetic content generated by advanced techniques like diffusion-based models.

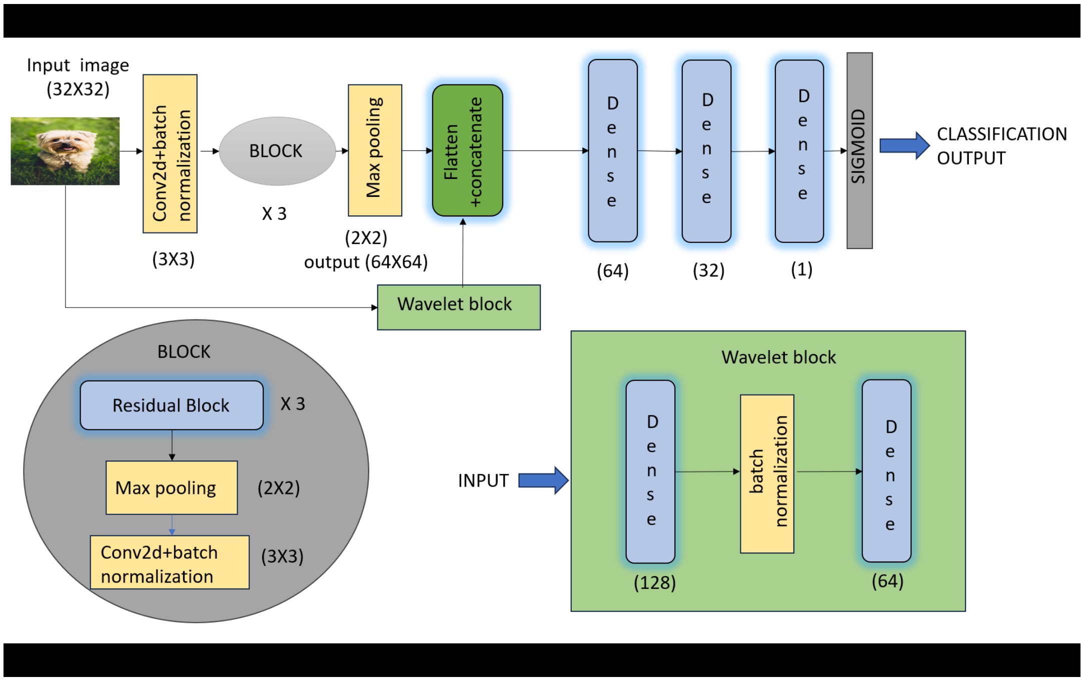


**Figure 2: Hybrid Model Architecture**

## 1.1 Wavelet Transform for Feature Extraction

Wavelet transforms play a crucial role in extracting features from the frequency domain. They were chosen because of their ability to capture both spatial and frequency information simultaneously. Unlike traditional Fourier transforms, wavelets allow multi-resolution analysis, making them particularly effective in identifying subtle, localized artifacts introduced by generative models.

Additionally, wavelet transforms are well-suited for grayscale images, which help reduce model complexity by lowering the number of parameters. These transforms are especially effective in identifying smoothing, sharpening, or other unnatural patterns often introduced by GANs and diffusion models.

For this approach, each grayscale image underwent decomposition using a biorthogonal wavelet transform (bior1.3). This process divided the image into four sub-bands: LL (low-frequency), LH, HL, and HH (high-frequency). These sub-bands captured various structural and texture details. The coefficients from these sub-bands were flattened and combined into a single feature vector. This wavelet-based feature representation complemented the spatial features extracted by the CNN, enabling a holistic approach to analyzing the image.

## 1.2 Hybrid Model Architecture

The proposed hybrid model incorporates two parallel branches: one focused on spatial feature extraction using a residual CNN, and the other dedicated to frequency feature extraction via wavelet transforms. These two branches process the input image in parallel, ensuring that both spatial and frequency-based artifacts are effectively captured. The features extracted from these branches are then merged and refined through fully connected layers, leading to the final classification output. Figure 2 explains how the model processes input data step by step

This dual-branch architecture is designed to utilize the strengths of both, frequency and the spatial domain aspects, allowing for robust detection of AI-generated images and their associated artifacts.

*1.2.1* ***Overview of Classifier Model Architecture.*** The proposed model begins by processing the input image through two branches:

A residual CNN for extracting spatial features. A wavelet-based branch for capturing frequency features. This combination ensures that the model can identify both low-level and high-level generative artifacts. The features from both branches are merged and further processed through fully connected layers, which generate the final classification. This multi-modal approach provides a comprehensive analysis of the image, enabling accurate identification of AI-generated content.

## 1.3 Training and Testing Procedure

The dataset used for training and testing consisted of two classes: REAL and FAKE. Images were preprocessed by normalizing pixel values to the range [0, 1]. A train-validation split was applied, and the model was trained using the following parameters which were obtained after experimentation:

| | |
|---|---|
| **Batch Size** | 128 |
| **Epochs** | 20 |
| **CallBacks** | EarlyStopping (patience=5)<br>ReduceLROnPlateaus (patience=3) |

**Table 1: Training Parameters**

# 2 TASK-2

## 2.1 Artifact Detection

The hybrid model proposed in Task-1 was able to effectively classify whether an image is real or AI-generated. In addition to this classification, the aim was to detect artifacts in the AI-generated images. For this, the **ViLT** model was used.
**Model used:** ViLT-B32-Finetuned-VQA.
**Model Source:** dandelin/vilt-b32-finetuned-vqa.
The ViLT model provides an important feature in pipeline: **artifact detection**. Once an image is identified as AI-generated, ViLT can detect artifacts which will then be fed to the Vision Language model to produce explanations. This ability to explain why an image is considered "AI-generated" helps in understanding the subtle visual cues that are often invisible to the human eye but detectable by the model. The AI-generated image, full list of artifacts along with a prompt of the form

***"Does this image have {artifact}?"***
was given to the model, to check which artifacts are present in the image. Those artifacts predicted with confidence 0.8 were chosen for each image. A maximum of 15 artifacts with the highest confidence levels were chosen and sent to the MiniCPM model.

*2.1.1* ***Model Architecture.*** The ViLT (Vision-and-Language Transformer) is specifically designed to perform tasks that involve both vision and language. It combines image understanding and textual reasoning for tasks like Visual Question Answering (VQA) [32]. The model we chose was pre-trained on VQA tasks. We further tried to fine-tune it with generated custom dataset of AI-generated images. This dataset was generated using various AI image generators, including Adobe Firefly, Standard Diffusion Models, and GigaGANs. We used LANCZOS [46] interpolation for downsampling the images in the dataset. Given that the dataset we generated was limited, the results of the finetuned models were not desirable(as shown in table 9). Hence the ViLT VQA base model was chosen, as shown in table 6

## 2.2 Explainability

We used SHAP [34] and Grad-CAM [33] to visualize the model's learnings. Figure 4 shows results of Grad-CAM on CIFAKE dataset. SHAP results are attached in the appendix.

*2.2.1* ***Grad-CAM.*** Gradient-weighted Class Activation Mapping (Grad-CAM), uses the gradients of any target concept, flowing into the final convolutional layer to produce a coarse localization map highlighting important regions in the image for predicting the concept. This is a technique for producing "visual explanations" for decisions from a large class of CNN-based models, making them more transparent. We implemented this from scratch and also divided the image into 3x3 regions to find the most likely position of the artifacts.

(1) *Setup:* Required libraries (tensorflow, numpy, matplotlib, cv2) were imported. The target convolutional layer ( usually the last layer (conv2d_20)) was selected.
(2) *Model Preparation:* Next, we defined a model (grad_model) that outputs both the feature maps of the target layer and the final predictions.

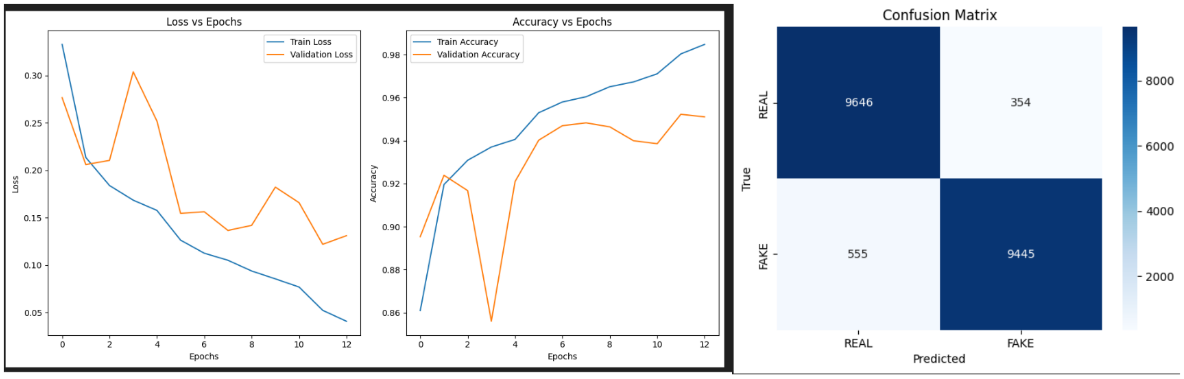


**Figure 3: (a) Loss vs Epochs and Accuracy vs Epochs curves (b) Confusion Matrix for CIFake Test Dataset**

(3) *Gradient Calculation:* tf.GradientTape was used to compute the gradients of the predicted output with respect to the target convolutional layer's activations.

(4) *Compute Weights:* Next, we applied global average pooling on the gradients to calculate weights for each channel in the feature map.

(5) *Generate Grad-CAM Heatmap:* A weighted combination of the feature map channels using the calculated weights was computed. ReLU was applied to retain positive contributions and normalize the heatmap. The heatmap was resized to the same dimensions as the input image using cv2.resize.

(6) *Position Analysis:* The heatmap was divided into a 3x3 grid corresponding to spatial regions: top-left, top-center, top-right, middle-left, middle-center, middle-right, bottom-left, bottom-center and bottom-right. The cumulative importance for each region was calculated by summing the heatmap values within that region. The region with the highest importance was identified.

(7) *Visualization:* The Grad-CAM heatmap was overlayed on the original image for visualization. The most important region in the image title was displayed and printed.

## 2.3 Visual Language Model (ViLM)

After detecting artifacts in the predicted AI-generated images, a ViLM model [42] was used to generate explanations for the presence of each artifact in the image.
**Model used:** MiniCPM-V-2 [9].
**Model source:** openbmb/MiniCPM-V-2

Before sending the AI-generated images to the ViLM, we upsampled the images to 128x128 using pre-trained **EDSR** (Enhanced Deep Residual Networks for Single Image Super-Resolution) [3] model. EDSR model is used for super-resolution (low-resolution image is converted to high-resolution image with enhanced features). The AI-generated image, list of artifacts in that image along with a prompt of the form
***"Describe how and where is {artifact} present in the image?"***
was given to the model, to generate explanations for each artifact.

## 3 DATA GENERATION METHOD

For further finetuning the ViLT model, images were generated using image generation tools like Adobe Firefly, Stable Diffusion, GANs, etc.

To generate the dataset, we provided a small set of artifact names as input to an AI model. The AI system then generated a total of 40 images based on these artifact prompts.

Each of the 40 generated images was associated with one or more of the artifacts from the list, and this relationship was captured by labeling each image in the dataset. Specifically, for each image, a binary "Yes/No" answer was provided to indicate whether a particular artifact was present. For example, if an image exhibited an inconsistent shadow or blurry edges, it would be labeled with "Yes" for those artifacts, while images that did not contain the specified artifact would be labeled "No".

This process helped create a diverse set of images where each image could exhibit different artifacts based on the input list.

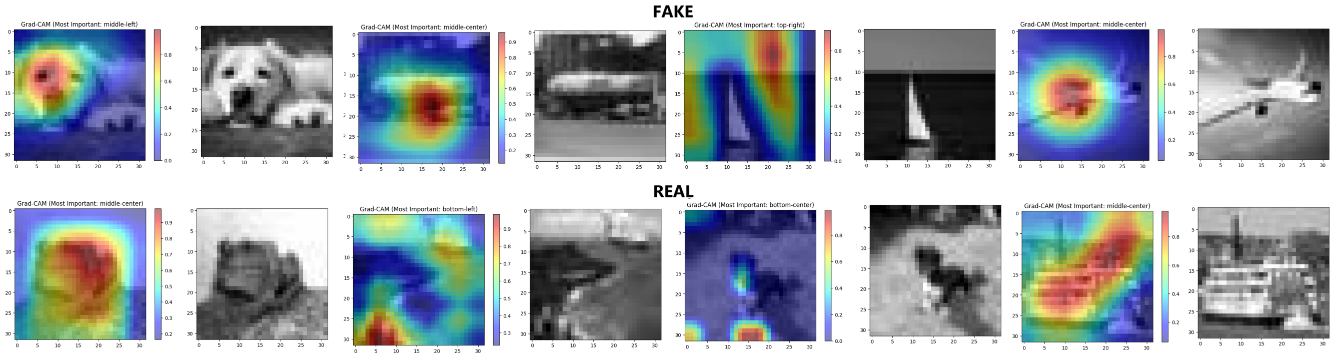


**Figure 4: Grad-CAM of some "FAKE" images in the first row and "REAL" in the second row. Red indicates the region of features which contributed the most to determining the label.**

# Results

## 1 TASK-1

### 1.1 Experimentation Results

During training, the hybrid model demonstrated a consistent improvement in accuracy and loss on both the training and validation sets. The early stopping callback ensured the model converged optimally.

These results highlight the model's ability to generalize effectively to unseen data within the same distribution.

### 1.2 Testing on CIFake and CIFar Test Datasets

To evaluate the model's generalizability to diverse generative techniques, the CIFake and CIFar test datasets were used.

The CIFake dataset consists of 10K test images. It also consists of 50K train images which we split into 80% training and 20% validation images.

After evaluation on CIFake, the hybrid model achieved a test accuracy of **95.11%**, demonstrating its robustness in identifying AI generated media across different generative methods. And for the CIFar(dataset of all real images) it sucessfully detected them with an accuracy of **93.21%**.

*1.2.1* ***Performance Metrics.*** The training curves and confusion matrix for the CIFake test dataset is presented in Figure 3, while the detailed classification report is shown in Table 2.

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
|---|---|---|---|---|
| REAL (0) | 0.95 | 0.96 | 0.96 | 10000 |
| FAKE (1) | 0.96 | 0.94 | 0.95 | 10000 |
| **Accuracy** | **0.95 (20000)** | | | |
| **Macro Avg** | 0.95 | 0.95 | 0.95 | 20000 |
| **Weighted Avg** | 0.95 | 0.95 | 0.95 | 20000 |

**Table 2: Classification Report for CIFake Test Dataset**

Tables 3 and 4 show the comparison between the proposed hybrid model and other popular models like MobileNet, VGG16, ResNet50 and ViT.

The proposed model has the perfect balance between accuracy and model size, achieving a remarkable 95% accuracy while being exceptionally lightweight with only 1.5 million parameters (6 MB). Compared to other models like ViT and VGG16, which require over 85 million and 29 million parameters respectively, proposed model demonstrates superior efficiency without compromising performance. MobileNet, while smaller in size, falls short in accuracy at 87%. Meanwhile, larger models like ResNet50, 23.5 million parameters, deliver significantly poorer results, achieving only 50% accuracy. This makes the proposed model an optimal choice for high-accuracy tasks on resource-constrained devices.

| Model Name | Total Params | Trainable Params | Non-trainable Params | Inference Time |
|---|---|---|---|---|
| **Proposed Model** | 1,573,633 (6 MB) | 1,570,241 | 3,392 | 0.071s |
| **MobileNet** | 6,906,053 (26.34 MB) | 1,575,937 | 2,178,240 | 0.082s |
| **VGG16** | 29,663,049 (113.16 MB) | 7,342,593 | 7,635,264 | 0.229s |
| **ResNet50** | 23,510,081 (78 MB) | 14,966,785 | 8,543,296 | 0.119s |
| **ViT** | 85,800,194 (327.3 MB) | 85,800,194 | 0 | 0.391s |

**Table 3: Comparing parameters and inference time for single image with other popular models (Task-1)**

| Model Name | Precision | Recall | F1-Score | Accuracy |
|---|---|---|---|---|
| **Proposed Model** | 0.95 | 0.95 | 0.95 | 0.95 |
| **MobileNet** | 0.87 | 0.87 | 0.87 | 0.87 |
| **VGG16** | 0.94 | 0.94 | 0.94 | 0.94 |
| **ResNet50** | 0.25 | 0.50 | 0.33 | 0.50 |
| **ViT** | 0.98 | 0.98 | 0.98 | 0.98 |

**Table 4: Comparing precision, recall, f1-score and accuracy with other popular models on the test data (Task-1)**

# 2 TASK-2

## 2.1 Artifact Detection

We tried to fine-tune the pre-trained ViLT model using a custom dataset of AI-generated images, but went on with the ViLT-finetuned-on VQA(base model) itself as it gave better results.

## 2.2 Example

Here is an example of how the model detects artifacts and generate explanations:
The image 5 was generated using adobe firefly by us. The proposed hybrid model detects Figure 5 as an AI-generated image. The corresponding Grad-CAM heatmap is shown in Figure 6

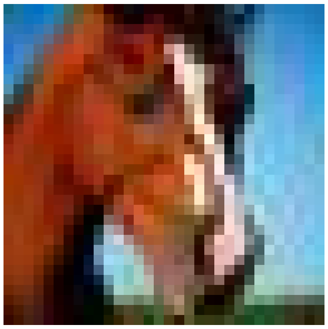


**Figure 5: Example image**

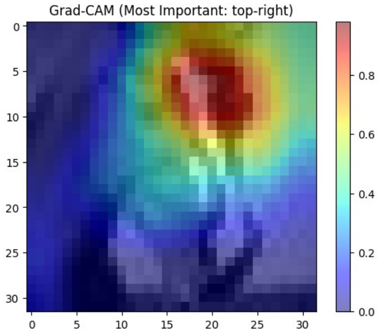


**Figure 6: Grad-CAM output**

ViLT model gives the list of artifacts (along with their confidence levels) in the AI-generated image. This list along with the image upsampled using EDSR model is sent to the ViLM (MiniCPM) which generates explanation for each artifact. Details below:

- **Unrealistic specular highlights: Confidence 0.9957:** Unrealistic specular highlights are present in the image due to a high level of noise or distortion, which affects how light interacts with objects and creates reflections.
- **Inconsistent shadow directions: Confidence 0.9943:** The inconsistent shadow directions are present in the horse's face, where some parts of its features have darker shadows while others do not.
- **Unnatural Lighting Gradients: Confidence 0.9902:** The Unnatural Lighting Gradients are present in the

| | Proposed Hybrid Model | ViLT | MiniCPM-V-2 | EDSR |
|---|---|---|---|---|
| **Total Parameters** | 1,573,633 (6 MB) | 112,782,338 | 3,430,000,000 | 12,000,000 |
| **Trainable Parameters** | 1,570,241 (5.99 MB) | 112,782,338 | 3,430,000,000 | 12,000,000 |
| **Non-trainable Parameters** | 3,392 (13.25 KB) | 0 | 0 | 0 |
| **Training Accuracy** | 0.9807 | 0.9486 | - | - |
| **Validation Accuracy** | 0.9497 | 0.8955 | - | - |
| **Training Loss** | 0.1548 | 0.1298 | - | - |
| **Validation Loss** | 0.2535 | 0.2132 | - | - |

**Table 5: Summary of all models used. The total parameters summed across all models is approximately 3.55 Billion**

image as a result of an artistic filter applied to create visual effects.

- **Dramatic lighting that defies natural physics: Confidence 0.9845:** The blue and white horse's face is illuminated with a bright light, which defies natural physics as it appears to be an artificial source of illumination.
- **Artificial smoothness: Confidence 0.9763:** Artificial smoothness is present in the image due to a blur effect applied, which smoothens out details and makes it appear less sharp.
- **Glow or light bleed around object boundaries: Confidence 0.9756:** The Glow or light bleed around object boundaries is present in the image, specifically on and near objects like horses.
- **Inconsistent material properties: Confidence 0.9714:** The Inconsistent material properties are present in the form of a blue horse with white stripes, which is not typical for horses.
- **Over-smoothing of natural textures: Confidence 0.9522:** Over-smoothing of natural textures is present in the image due to a lack of sharpness and detail, which can be observed as an overall blurriness or softening effect on various elements within it.
- **Movie-poster like composition of ordinary scenes: Confidence 0.9462:** The image has a movie-poster like composition of ordinary scenes, as it features an abstract representation with colors and shapes that resemble the elements commonly found in such compositions.
- **Synthetic material appearance: Confidence 0.9280:** The presence of Synthetic material appears as a blur or distortion in the image, possibly due to digital manipulation.
- **Scale inconsistencies within single objects: Confidence 0.9202:** Scale inconsistencies within single objects are present in the image due to digital noise or blurriness, which affects different parts of an object differently.

Table 5 we can infer that proposed model is relatively light when compared with other architectures and can be run even on smaller devices like mobile phones.

Table 6 shows comparison between the proposed model and other popular MultiModals for artifact detection , ViLT base VQA model gives similar results given by large multimodals like GPT-4 and Gemini.

Table 7 shows the comparison between MiniCPM and Llava model, MiniCPM being much smaller gives more accurate explanations, also having lower inference time which makes it faster and compatible with small end-to-end devices.

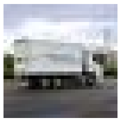
| Image | GPT-4 | Gemini | ViLT fine-tuned on VQA |
|---|---|---|---|
| | Inconsistent object boundaries | Inconsistent object boundaries | Inconsistent object boundaries: 0.9843* |
| | Unrealistic specular highlights | Discontinuous Surfaces | Unrealistic specular highlights: 0.9966 |
| | Inconsistent material properties | Non-manifold Geometries in Rigid Structures | Inconsistent material properties: 0.6904 |
| | Inconsistent shadow directions | Aliasing along High-contrast Edges | Inconsistent shadow directions: 0.9948 |
| | | Jagged Edges in Curved Structures | Discontinuous surfaces 0.9923 |
| | Improper Fur Direction Flows | Inconsistent Object Boundaries | Improper Fur Direction Flows: 0.9903 |
| | Texture Bleeding Between Adjacent Regions | Inconsistent Shadow Directions | Texture Bleeding Between Adjacent Regions: 0.9812 |
| | Over-Smoothing of Natural Textures | Misshapen Ears or Appendages | Over-Smoothing of Natural Textures 0.9900 |
| | Unrealistic Eye Reflections | Over-Smoothing of Natural Textures | Unrealistic Eye Reflections: 0.9950 |
| | Misshapen Ears or Appendages | Unrealistic Specular Highlights | Misshapen Ears or Appendages: 0.9858 |
| | Inconsistent Shadow Directions | Improper Fur Direction Flows | Inconsistent Shadow Directions: 0.9999 |
| | Unnatural Lighting Gradients | Unrealistic Eye Reflections | Unnatural Lighting Gradients: 0.9985 |
| | Depth Perception Anomalies | | Depth Perception Anomalies: 0.9971 |
| | Artificial Smoothness | | Artificial Smoothness: 0.9992 |
| | Exaggerated Characteristic Features | | Unrealistic Specular Highlights: 0.9999 |
| | | | Inconsistent Object Boundaries: 0.9999 |
|  | Inconsistent shadow directions | Inconsistent object boundaries | Depth perception anomalies: 0.8662 |
| | Depth perception anomalies | Discontinuous surfaces | Unnatural Lighting Gradients: 0.8328 |
| | Unrealistic specular highlights | Unrealistic specular highlights | Discontinuous surfaces: 0.9933 |
| | Inconsistent material properties | Unnatural Lighting Gradients | Artificial smoothness: 0.9628 |
| | Unnatural lighting gradients | | Unrealistic specular highlights: 0.9953 |
| | Artificial smoothness | | Inconsistent material properties: 0.7316 |
| | Texture repetition patterns | | Inconsistent shadow directions: 0.9934 |
| | | | Inconsistent object boundaries: 0.8463 |

**Table 6: Comparison between model (ViLT base VQA) and other popular Multimodals. *The score such as 0.9843 denotes the confidence of the model in detecting the given artifact.**

| Artifact | MiniCPM-V-2 (3.43B) | LLAVA (34B) |
|---|---|---|
| **Inconsistent shadow directions** | The inconsistent shadow directions are present in the horse's face, where some parts of its features have darker shadows while others do not. | The image appears to have inconsistent shadow directions, which can be observed in the region where the blue object meets the yellow background. The shadows cast by the blue object onto the yellow background seem to be in different directions, which is unusual and suggests that the image may have been manipulated or that the lighting conditions are not consistent. |
| **Unnatural Lighting Gradients** | The Unnatural Lighting Gradients are present in the image as a result of an artistic filter applied to create visual effects. | The image shows a close-up of a person wearing a blue garment, possibly a jacket or a shirt. The lighting gradient is unnatural because it appears to be a digital manipulation or an artistic effect rather than a natural lighting condition. The gradient is present across the image, with a sharp transition from a bright yellow to a darker blue, which is not a typical lighting scenario. This gradient is likely added to the image to create a specific mood or aesthetic, rather than reflecting the actual lighting conditions of the |
| **Synthetic material appearance** | The presence of Synthetic material appears as a blur or distortion in the image, possibly due to digital manipulation. | The image appears to be a close-up of a person wearing a blue garment, possibly a jacket or a shirt, with a white stripe or detail on the right side. The synthetic material appearance is likely due to the texture and sheen of the fabric, which can be seen in the close-up view. The material looks smooth and possibly has a slight reflection, which is common in synthetic fabrics. The exact location of the synthetic material is not clear due to the close-up nature of the |

**Table 7: Comparison of MiniCPM-V-2 model (we're using) output with LLava Model on the image 5. with maximum token 50 as argument**

# Discussion

## 1 TASK-1

### 1.1 Adversarial Training

This implementation builds upon the previously proposed hybrid model using adversarial training [49] to enhance its robustness against adversarial attacks. Using the Fast Gradient Sign Method (FGSM) [48] , perturbations are introduced to input images, creating adversarial examples. These perturbations are designed to be nearly invisible to the human eye but effective in misleading the model. By exposing the model to these distorted samples during training, it learns to better recognize and resist such attacks. This approach has made the model robust to the adversarial attacks on the CIFAKE test set with a test accuracy of 94.27%.

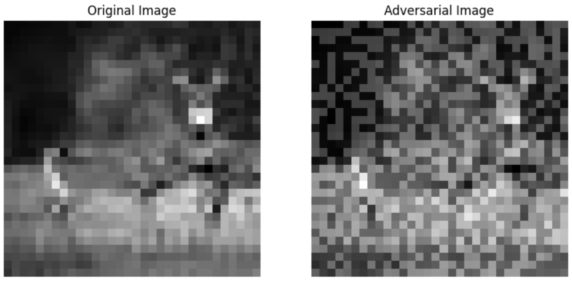


**Figure 7: Adversarial Training**

Moreover, the inclusion of wavelet features alongside adversarial training allows the model to capture both image details and the unique characteristics of adversarial perturbations. This combined strategy not only boosts classification accuracy but also enhances the model's ability to generalize effectively to other datasets, such as CIFar-10. The test accuracy on CIFar-10 dataset was found to be 93.06%. As a result, the model proves to be more reliable and adaptable across a variety of challenging scenarios.

| - | Actual FAKE | Actual REAL |
|---|---|---|
| Predicted FAKE | 9380 | 620 |
| Predicted REAL | 527 | 9473 |

**Table 8: Confusion Matrix of CIFake**

## 1.2 Limitations, Other Experimentation, and Future Work

Despite the strong performance, certain limitations exist:

- The model is limited to grayscale inputs and low-resolution images (32 × 32) and particularly to CIFake and CIFar datasets.
- Further exploration of alternative wavelet families could improve feature extraction.
- Incorporating attention mechanisms may enhance the model's ability to focus on relevant regions of the image.
- **Generalizability:** The model's effectiveness on datasets with different content types, such as human faces, remains uncertain.
- **Downsampling Challenges:** Without knowledge of the original downsampling technique, reconstructing features for 32×32 predictions is inherently difficult.
- **Evolving Generative Models:** Rapid advancements in generative technologies could diminish the model's ability to detect unseen AI-generated content effectively.

*1.2.1* ***Other Experimentation.*** In addition to the custom hybrid model, experimentation was conducted using a Vision Transformer (ViT) [15] with 85 million parameters. The ViT achieved an impressive accuracy of 98% on the CIFake test dataset. However, the hybrid model used in this study has only 1.6 million parameters, making it significantly more efficient in terms of computational and memory requirements.

This trade-off between performance and complexity favors the custom hybrid model, especially for resource-constrained environments. For reference, the code implementation of the ViT will be attached for further exploration and replication.

# 2 TASK-2

## 2.1 Limitations of current solution.

(1) We use 3 separate models in the pipeline each of which are separately trained for their specific task.
(2) While generating the dataset of images with certain artifacts we can not say with certainty that only the given artifacts and no other artifacts were added by the AI.
(3) As there was no image description we could not train an embedding layer or use CLIP method which could establish a relationship between the visual and textual features.
(4) Converting visual elements into text may lose context, such as the spatial relationships between objects. Ensuring the generated text is grammatically correct and contextually coherent requires sophisticated language models.
(5) The proposed hybrid model has a small number of parameters for efficiency. However increasing the size may lead to better performance.
(6) Grad-CAM++ and integrated gradients are some other advanced techniques that may improve localization in the visualisation of the outputs.
(7) Since the model is trained on CIFAKE dataset which is based on the CIFAR dataset which has images from 2008 it may not perform well on recent image data.

## 2.2 Observations regarding the data.

(1) The given test dataset consisted of 32 x 32 sized downsampled images which makes it hard to identify artifacts and tell whether the image was AI-generated since all of them look blurry.
(2) For real world-applications it would be better to use higher resolution images with size 512 x 512.
(3) The CIFAKE dataset contained only 10 classes of images and did not include humans, thus a bigger and more diverse dataset is required.

## 2.3 Implementation difficulties.

.

(1) There were some issues with vanishing gradient leading to division by zero in the gradient computation in Grad-CAM. As a work-around, for such cases we use an earlier convolution layer such as conv2d_14 instead of the last layer conv2d_20.

(2) Grad-CAM had to be implemented from scratch as existing implementations were built for pre-trained models and implicitly required functions that were not present in our custom implementation of the proposed hybrid model for Task-1.

(3) Fine-tuning the ViLT on custom dataset led to poor performance with the model detecting every artifact from the given list of artifacts.

## 2.4 Potential improvements to the algorithm.

(1) It would be better to have one unified model instead of 3 separate models as propagating the loss through it would ensure all parameters are updated in sync with each other and not independently.

(2) Using dataset with human faces, and more diverse classes of data would improve the generalization of the model.

## 2.5 Broader applications of our approach in real-world scenarios.

(1) For deep-fake detection and automatic take-down of fake content on the internet and on social media platforms.

(2) To add explainability to current automated fake content detection systems so that we can trust and verify the reasons behind tagging content as fake.

# Appendix

## 1 TASK-1

### 1.1 Other Models

*1.1.1* ***Vision Transformer.*** We experimented with a pre-trained Vision transformer and fine-tuned it on the CIFAKE dataset. It achieved 98% accuracy, however this came at the cost of it being a large model with over 85 million parameters.

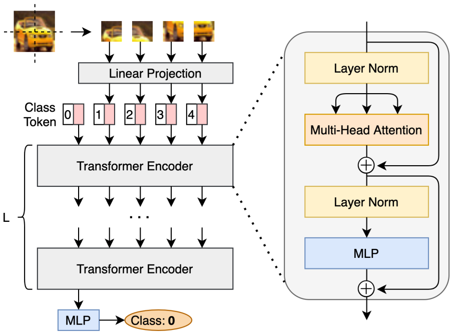


**Figure 8: ViT architecture.**

The architecture of the vision transformer mainly consists of:

(1) Patch Embedding layer which converts patches of the image into embeddings.
(2) A learnable positional embedding (pos embedding) is added to the patch embeddings. A classification token (cls token) is added at the start which acts as a kind of summary of the image.
(3) The attention layer multiplies the input with the weight matrices to generate Q, K, V tuples for each input. This generates a context aware representation for each image patch.
(4) Multi-head attention employs multiple sets of Q, K, V weight matrices. These run in parallel, each 'head' potentially focusing on different aspects of the input relationships. The outputs from each head are concatenated and linearly transformed, giving the model a richer representation of the input sequence.
(5) The output of the multi-head attention module and the subsequent 'Add and Norm' layer is fed into the feedforward layer of each transformer block.
(6) The feedforward layer consists of two linear transformations with a non-linear activation function in between to add further representational power to the model. It also contains dropout layers to reduce overfitting.
(7) Finally the classification is made.

We also experimented with pre-trained models of VGG16, MobileNet and ResNet50 and fine-tuned it on the CIFAKE dataset.

*1.1.2* ***VGG 16.*** [4] It is a deep convolutional neural network that consists of 16 layers including 13 convolutional layers and 3 fully connected layers, all using small 3x3 filters. It achieved an accuracy of 94%.

*1.1.3* ***MobileNet.*** [5] It is a lightweight convolutional neural network and uses depthwise separable convolutions, which reduces number of parameters and computational cost. It achieved an accuracy of 87%.

*1.1.4* ***ResNet50.*** [6] It is a deep residual neural network with 50 layers that uses skip connections and solves the problem of vanishing gradients. It achieved an accuracy of 50%.

## 2 TASK-2

### 2.1 Artifact Detection

*2.1.1* ***CLIP.*** [2] It is a model by OpenAI for contrastive learning, with the aim of aligning images and text in the same feature space. This enables it to be great in zero-shot learning. The model can classify images using their textual descriptions without requiring task-specific training data. CLIP is very powerful for tasks like image classification and retrieval but does not directly handle tasks like image captioning or question answering, which makes it less suitable for artifact detection in images where the language understanding has to be very precise.

*2.1.2* ***BLIP.*** [1] It is a model by Salesforce Research extends CLIP but provides a bootstrapping mechanism for better image-text alignment. It is particularly suitable for image captioning and visual question answering (VQA) [32]. Generating textual descriptions or answering questions based on images is useful for BLIP in strengthening the language-vision alignment. However, while being strong in caption

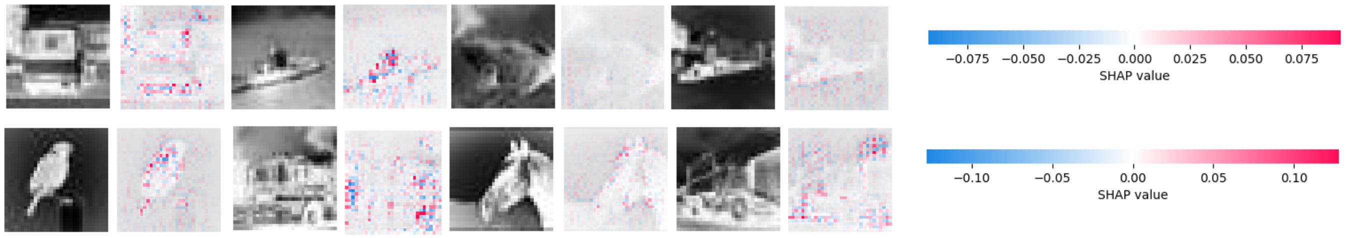


**Figure 9: Images with their corresponding SHAP outputs: "FAKE" images in the first row and "REAL" images in the second row.**

generation and VQA, it failed to perform at par with ViLT in our experiments for artifact detection.

## 2.2 Fine-Tuned Models (ViLT)

Fine tuned models: These models were finetuned on the dataset with 40 images with different methods of augmentation. The model_40_12 name represents that the dataset was finetuned on 40 images where the augmentation of the first label was performed until its count was equal to 1/2 of the second label. The same nomenclature follows for the other models. The training and validation results for each model are shown in 10.

In the given table, we have stored the results generated from each of the four finetuned models (40_12, 40_13, 40_14, 40_23) and are comparing them with the artifacts predicted by GPT4.

The image was given to each of the models, the code for which can be found in the folder
(path: final/other_models/fine_tuned_vilt_models/
<mode_you_want_import>/implement.ipynb). We have also showed how confident our model was in the prediction of each artifact for each image. The artifacts with confidence greater than 90% were chosen as the top artifacts. The corresponding results imply that the model improves with the provision of a more diverse dataset.

## 2.3 Explainability

*2.3.1* ***SHAP***. **(SHapley Additive exPlanations)** is a unified framework for interpreting predictions. SHAP assigns each feature an importance value for a particular prediction. However, its outputs are not at par with gradcam. Pink indicates features or regions contributing the most to the prediction of the predicted label and blue indicates regions contributing to the prediction of the other label. In many of the outputs we see a mix of blue and pink which is hard to interpret.

Figure 9 shows sample outputs from SHAP.

## 2.4 Visual Language Model

### *2.4.1* ***LLAVA***.

**Model source:** llava-v1.6-34b-hf

The LLaVA-NeXT model [47] was proposed in LLaVA-NeXT: Improved reasoning, OCR, and world knowledge by Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, Yong Jae Lee. LLaVa-NeXT (also called LLaVa-1.6) improves upon LLaVa by increasing the input image resolution and training on an improved visual instruction tuning dataset to improve OCR and common sense reasoning. LLaVa combines a pre-trained large language model with a pre-trained vision encoder for multimodal chatbot use cases. It has 34.8 billion parameters which makes it quite heavy compared to Mini-CPM which has only 3.43 billion parameters and imcompatible with low end devices like mobile phones.

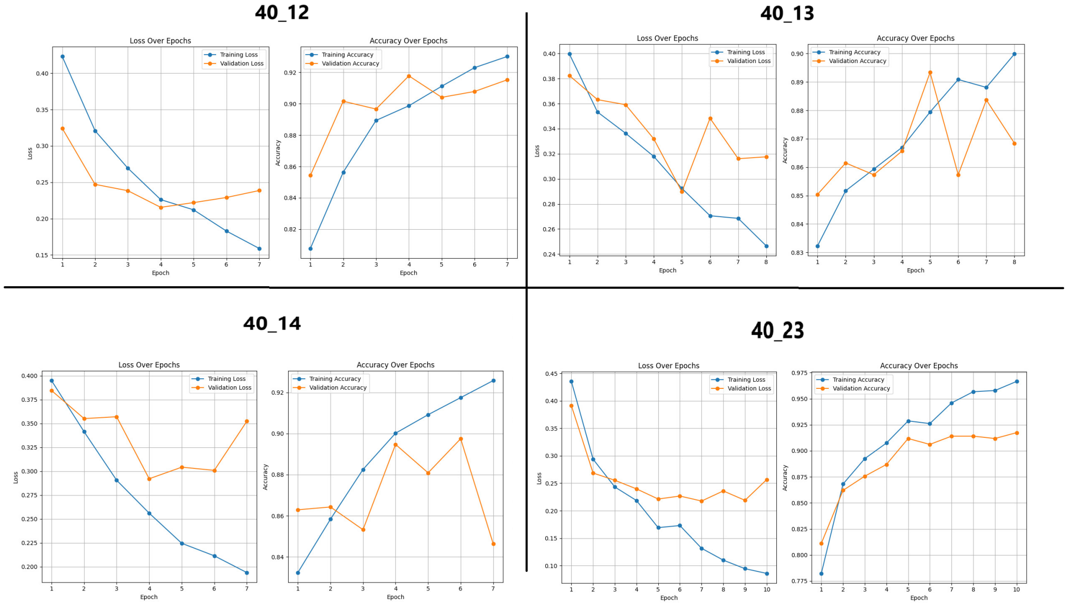


**Figure 10: Loss vs Epochs and Accuracy vs Epochs curves for differently fine-tuned ViLT models**

| ChatGPT | Model 40_14 | Model 40_13 | Model 40_12 | Model 40_23 |
|---|---|---|---|---|
| Inconsistent object boundaries | Inconsistent material properties,<br>Confidence: 0.8549 | Unrealistic specular highlights,<br>Confidence: 0.9973 | Over-smoothing of natural textures,<br>Confidence: 1.0000 | Unrealistic specular highlights,<br>Confidence: 1.0000 |
| Inconsistent shadow directions | Inconsistent shadow directions,<br>Confidence: 0.8355 | Inconsistent shadow directions,<br>Confidence: 0.9793 | Inconsistent shadow directions,<br>Confidence: 1.0000 | Incorrect reflection mapping,<br>Confidence: 0.9997 |
| Multiple light source conflicts | Discontinuous surfaces,<br>Confidence: 0.8051 | | Improper fur direction flows,<br>Confidence: 1.0000 | Depth perception anomalies,<br>Confidence: 0.9995 |
| Discontinuous surfaces | | | Texture bleeding between adjacent regions,<br>Confidence: 1.0000 | Inconsistent shadow directions,<br>Confidence: 0.9995 |
| Artificial smoothness | | | Depth perception anomalies,<br>Confidence: 0.9999 | Artificial smoothness,<br>Confidence: 0.9987 |

**Table 9: Output comparison of fine-tuned ViLT models with ChatGPT**

## REFERENCES

[1] Junnan Li, Dongxu Li, Caiming Xiong, Steven Hoi. *Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation.*

[2] Hong-You Chen, Zhengfeng Lai, Haotian Zhang, Xinze Wang, Marcin Eichner, Keen You, Meng Cao, Bowen Zhang, Yinfei Yang, Zhe Gan. *Contrastive Localized Language-Image Pre-Training.*

[3] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, Kyoung Mu Lee. *Enhanced Deep Residual Networks for Single Image Super-Resolution.*

[4] Karen Simonyan, Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition.*

[5] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.*

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. *Deep Residual Learning for Image Recognition.*

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Ilya Sutskever. *Learning Transferable Visual Models From Natural Language Supervision.* 26 Feb 2021.

[8] Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, Maosong Sun. *MiniCPM: Unveiling the Potential of Small Language Models with Scalable Training Strategies.* 3 Jun 2024.

[9] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, Guoyang Zeng, Dahai Li, Zhiyuan Liu, Maosong Sun. *MiniCPM-V: A GPT-4V Level MLLM on Your Phone.* 3 Aug 2024.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. *Harmonizing Visual and Textual Embeddings for Zero-Shot Text-to-Image Customization.* 21 Mar 2024.

[11] Saeid Asgari Taghanaki, Aliasghar Khani, Ali Saheb Pasand, Amir Khasahmadi, Aditya Sanghi, Karl D.D. Willis, Ali Mahdavi-Amiri. *TExplain: Explaining Learned Visual Features via Pre-trained (Frozen) Language Models.* 2 May 2024.

[12] Saeid Asgari Taghanaki, Aliasghar Khani, Ali Saheb Pasand, Amir Khasahmadi, Aditya Sanghi, Karl D.D. Willis, Ali Mahdavi-Amiri. *LLaMA: Open and Efficient Foundation Language Models.* 27 Feb 2023.

[13] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, Karen Simonyan. *Flamingo: a Visual Language Model for Few-Shot Learning.* 15 Nov 2022.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. *Deep Residual Learning for Image Recognition.*

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.* 3 Jun 2021.

[16] Huanrui Yang, Hongxu Yin, Maying Shen, Pavlo Molchanov, Hai Li, Jan Kautz. *Global Vision Transformer Pruning with Hessian-Aware Saliency.* 29 Mar 2023.

[17] Hangbo Bao, Li Dong, Songhao Piao, Furu Wei. *BEiT: BERT Pre-Training of Image Transformers.* 3 Sep 2022.

[18] Chengyue Gong, Dilin Wang, Meng Li, Vikas Chandra, Qiang Liu. *Vision Transformers with Patch Diversification.* 11 Jun 2021.

[19] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, Lucas Beyer. *How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers.* 23 Jun 2022.

[20] Xiangning Chen, Cho-Jui Hsieh, Boqing Gong. *When Vision Transformers Outperform ResNets without Pre-training or Strong Data Augmentations.* 13 Mar 2022.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. *Scaling Vision with Sparse Mixture of Experts.*

[22] Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, Neil Houlsby. *Deep Residual Learning for Image Recognition.* 10 Jun 2021.

[23] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, Shuicheng Yan. *Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet.* 30 Nov 2021.

[24] Yulin Wang, Rui Huang, Shiji Song, Zeyi Huang, Gao Huang. *Not All Images are Worth 16x16 Words: Dynamic Transformers for Efficient Image Recognition.* 26 Oct 2021.

[25] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, Baining Guo. *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows.*

[26] Chen Zhu, Wei Ping, Chaowei Xiao, Mohammad Shoeybi, Tom Goldstein, Anima Anandkumar, Bryan Catanzaro. *Long-Short Transformer: Efficient Transformers for Language and Vision.* 7 Dec 2021.

[27] Pichao Wang, Xue Wang, Fan Wang, Ming Lin, Shuning Chang, Hao Li, Rong Jin. *KVT: k-NN Attention for Boosting Vision Transformers.* 22 Jul 2022.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. *Deep Residual Learning for Image Recognition.*

[29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. *Deep Residual Learning for Image Recognition.*

[30] Nikola Popovic, Danda Pani Paudel, Thomas Probst, Luc Van Gool. *Improving the Behaviour of Vision Transformers with Token-consistent Stochastic Layers.* 14 Jul 2022.

[31] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, Devi Parikh. *VQA: Visual Question Answering.* 27 Oct 2016.

[32] Ana Cláudia Akemi Matsuki de Faria, Felype de Castro Bastos, José Victor Nogueira Alves da Silva, Vitor Lopes Fabris, Valeska de Sousa Uchoa, Décio Gonçalves de Aguiar Neto, Claudio Filipi Goncalves

dos Santos. *Visual Question Answering: A Survey on Techniques and Common Trends in Recent Literature.* 2 Jun 2023.

[33] Scott Lundberg, Su-In Lee *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization.* 3 Dec 2019

[34] Ana Cláudia Akemi Matsuki de Faria, Felype de Castro Bastos, José Victor Nogueira Alves da Silva, Vitor Lopes Fabris, Valeska de Sousa Uchoa, Décio Gonçalves de Aguiar Neto, Claudio Filipi Goncalves dos Santos. *A Unified Approach to Interpreting Model Predictions* .25 Nov 2017.

[35] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, Vineeth N Balasubramanian *Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks* .25 Nov 2017.

[36] Soumick Chatterjee, Arnab Das, Chirag Mandal, Budhaditya Mukhopadhyay, Manish Vipinraj, Aniruddh Shukla, Rajatha Nagaraja Rao, Chompunuch Sarasaen, Oliver Speck, Andreas Nürnberger *Torch-Esegeta: Framework for Interpretability and Explainability of Image-based Deep Learning Models* .7 Feb 2022.

[37] Quoc Hung Cao, Truong Thanh Hung Nguyen, Vo Thanh Khang Nguyen, Xuan Phong Nguyen *A Novel Explainable Artificial Intelligence Model in Image Classification problem* .9 Jul 2023.

[38] Luca Longo, Mario Brcic, Federico Cabitza, Jaesik Choi, Roberto Confalonieri, Javier Del Ser, Riccardo Guidotti, Yoichi Hayashi, Francisco Herrera, Andreas Holzinger, Richard Jiang, Hassan Khosravi, Freddy Lecue, Gianclaudio Malgieri, Andrés Páez, Wojciech Samek, Johannes Schneider, Timo Speith, Simone Stumpf *Explainable Artificial Intelligence (XAI) 2.0: A Manifesto of Open Challenges and Interdisciplinary Research Directions.*30 Jan 2022.

[39] Prashant Gohel, Priyanka Singh, Manoranjan Mohanty *Explainable AI: current status and future directions.*30 Oct 2019.

[40] PAshish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin *Attention Is All You Need* .30 Oct 2017.

[41] Weibing Zhao *Block-Diagonal Guided DBSCAN Clustering* .27 Apr 2024.

[42] Florian Bordes, Richard Yuanzhe Pang, Anurag Ajay, Alexander C. Li, Adrien Bardes, Suzanne Petryk, Oscar Mañas, Zhiqiu Lin, Anas Mahmoud, Bargav Jayaraman, Mark Ibrahim, Melissa Hall, Yunyang Xiong, Jonathan Lebensold, Candace Ross, Srihari Jayakumar, Chuan Guo, Diane Bouchacourt, Haider Al-Tahan, Karthik Padthe, Vasu Sharma, Hu Xu, Xiaoqing Ellen Tan, Megan Richards, Samuel Lavoie, Pietro Astolfi, Reyhane Askari Hemmat, Jun Chen, Kushal Tirumala, Rim Assouel, Mazda Moayeri, Arjang Talattof, Kamalika Chaudhuri, Zechun Liu, Xilun Chen, Quentin Garrido, Karen Ullrich, Aishwarya Agrawal, Kate Saenko, Asli Celikyilmaz, Vikas Chandra *An Introduction to Vision-Language Modeling* .27 May 2024.

[43] Alexander B. Gurvich, Aaron M. Geller *Firefly: a browser-based interactive 3D data visualization tool for millions of data points*

[44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, Björn Ommer *High-Resolution Image Synthesis with Latent Diffusion* Models

[45] Shaoqin Pan1*, Yanhong Ma2,a, Zhenghan Chen1,c *A Study of Midjourney-based Artificial Intelligence in Clothing Design Innovation*

[46] Jorge Garza-Vargas Archit Kulkarni. *The Lanczos Algorithm Under Few Iterations: Concentration and Location of the Output*

[47] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, Chunyuan Li. *LLaVA-NeXT-Interleave: Tackling Multi-image, Video, and 3D in Large Multimodal Models*

[48] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy. *Explaining and Harnessing Adversarial Examples*

[49] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy. *Adversarial Training: A Survey*