Deep Learning in Cartoon Moderation: Distinguishing Child-Friendly Content with CNN Architectures

S Kumar Reddy Mallidi(✉), Sujana Bellapukonda, Nikhitha Gollapudi, Karthik Malaka, P. Sreeshanth, and Sai Sri Harsha Polisetti

Computer Science and Engineering, Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh 534101, India
satya.cnis@gmail.com

Abstract. This study introduces a deep learning solution to safeguard children from potentially harmful content in cartoons, such as violence and adulterous scenes. Recognizing the possible negative influence of such content on young minds, we developed a hybrid Convolutional Neural Network (CNN) model that combines the strengths of InceptionV3 and VGG16. This model is meticulously designed to effectively identify and filter out aggressive and inappropriate content. Our comparative analysis highlights the proposed model's superior performance, including established models like VGG16, VGG19, ResNet50, and InceptionV3. It is particularly notable for its high efficiency and fewer parameters, successfully addressing the common issue of overfitting often encountered in deep learning models. This advancement in performance is critical for the model's ability to identify and mitigate exposure to harmful content in cartoons accurately. The results of this research represent a significant advancement in content moderation technology for children's media, paving the way for a safer and more suitable viewing environment.

Keywords: Deep Learning · Convolutional Neural Networks (CNNs) · Transfer Learning · Child Safety · Content Moderation

1 Introduction

Cartoons, which serve as a primary source of entertainment for many children, occasionally include scenes that may not be suitable for a young audience. Considering the profound impact these animations can have on a child's development and worldview, it becomes imperative to identify and mitigate content that could negatively influence their understanding and behavior. This research is dedicated to developing a system that can effectively detect and filter out violent and inappropriate content from cartoons, enhancing their suitability for younger viewers.

This study harnesses the power of Deep Learning, known for its exceptional ability to process and analyze unstructured data such as images and videos. In an age where

P. Pareek et al. (Eds.): IC4S 2024, LNICST 597, pp. 241–256, 2025.
https://doi.org/10.1007/978-3-031-77075-3_20

digital media is omnipresent, addressing the more subtle and often overlooked aspects of popular content is increasingly important. Our methodology employs deep learning algorithms, modeled after the complexity of human neural networks, to identify and interpret patterns that signal violent or otherwise harmful content. The objective is to provide children with a safer and more appropriate viewing experience, ensuring that the cartoons they watch contribute positively to their growth and learning.

1.1 Literature Review

The exploration of violence detection in cartoons and animated content has become increasingly prominent with the integration of advanced machine-learning techniques. Early studies such as those by Khalil et al. [1] and Khan et al. [2] employed visual features and multimedia content detection techniques to identify violent cartoon scenes. These foundational works paved the way for more advanced approaches. However, despite significant progress in violence detection, the specific classification of anime, particularly in terms of adult content, remains relatively unexplored.

The shift to deep learning marked a turning point in this field. Khan et al. [3] showcased the effectiveness of the MobileNet architecture in categorizing violent content in cartoons. Similarly, Taha et al. [4] introduced InspectorNet, a transformer network for violence detection in animated content. The use of pre-trained models such as VGG16, VGG19, ResNet50, and InceptionV3 in studies like those by Honarjoo et al. [5], Sumon et al. [6], and Jain and Vishwakarma [7] further illustrates the potential of leveraging existing architectures. Notably, Arun Akash et al. [8] explored human violence detection using deep learning techniques, indicating the versatility of these models in different contexts.

The broader field of video classification and violence detection, including works like Bermejo et al. [9], Bin Jahlan and Elrefaei [10], and Yousaf and Nawaz [11], has provided valuable insights into various techniques for identifying violent and inappropriate content. Additionally, creating specialized datasets, as demonstrated by Zhou et al. [12] with ToonNet, is crucial for model training in specific domains like cartoons. A comprehensive review of automatic video classification by Rani et al. [13] further underscores this area's diverse methodologies and challenges, highlighting the importance of tailored approaches for different video genres, including anime.

The specific challenge of classifying anime content, especially in detecting adult content, is a gap in the current research landscape. While existing studies have significantly advanced violence detection in cartoons, they generally do not address the unique aspects of anime. This gap presents an opportunity for future research, especially considering anime's growing popularity. The work of Singh and Tyagi [14] and Ding et al. [15], which explore computational comparisons and 3D convolutional neural networks in violence detection, could be adapted for anime classification. Additionally, the enhancements in violence detection techniques, such as those by Abdelkader and Abbass [16], suggest a trend towards more sophisticated models that could be pivotal in anime classification.

1.2 Contribution of This Work

This study unveils an innovative deep-learning model designed to excel beyond current CNN architectures in accurately differentiating between child-friendly and non-friendly content. The contributions of this research are varied and significant.

Innovative Model Design. We introduce a CNN model that effectively combines the attributes of InceptionV3 and VGG16. This results in a less complex and more resource-efficient model than existing architectures such as VGG16, VGG19, ResNet50, and InceptionV3. The design optimizes for efficiency without compromising the ability to process complex content patterns.

Comparative Analysis. Through thorough comparative analysis, the model has demonstrated superior performance over traditional architectures in accurately classifying content. A key aspect of this enhanced performance is its ability to prevent overfitting, a common challenge in deep learning models, thereby ensuring more reliable and robust content categorization.

Practical Implications. The model's efficacy and efficiency carry significant real-world applications. Most notably, its potential to improve content moderation algorithms stands out, particularly for children's media. By providing more accurate content filtering, this model represents a substantial step forward in creating safer digital environments for children.

Overall, this paper presents a leap forward in utilizing deep learning for content moderation, showcasing a model that performs with high accuracy and addresses critical challenges in the field of media safety for younger audiences.

2 Related Work

Deep learning has significantly transformed the realm of image recognition, particularly through the advancement of Convolutional Neural Networks (CNNs) such as VGG16 and ResNet50. These models have become benchmarks in the field, establishing new standards for classification systems even for applications like violence detection from images and videos [17, 18]. They are distinguished by their unique architectures and their implications on the accuracy and efficiency of image classification tasks. Their innovative design and proven effectiveness have made them foundational models in both academic research and practical applications in image recognition.

2.1 VGG-16

VGG-16 is a prominent model in deep learning, known for its 16-layer deep convolutional neural network architecture. VGG-16 is well-known for its capacity to categorize images into numerous classes using a structure that consists of compact, closely arranged filters, as illustrated in Fig. 1. This architecture is crucial for efficiently acquiring diverse image characteristics. VGG16 demonstrates superior performance in robust object recognition because of its training on an extensive dataset of images. The model's extensive parameter count of 31.6 million requires significant computational and memory resources. The model's design has been popular in object classification but is facing hurdles. The complexity of its architecture presents challenges in transfer learning and interpretability, which are crucial factors in its application.

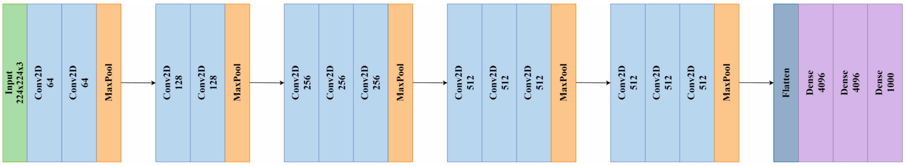

Fig. 1. VGG16: 16-layer network excelling in image recognition.

2.2 VGG-19

VGG-19 is known for its 19-layer design, which employs a simple yet powerful method for image identification. The network's architecture, depicted in Fig. 2, is specifically crafted to systematically learn and identify patterns in images via a progressive layering approach. The intentional layer arrangement improves the model's ability to extract detailed features, hence greatly enhancing its overall performance in picture categorization.

VGG-19's design is deep, resulting in a large parameter count of around 171 million. Having a large number of parameters can be limiting, especially in situations that need real-time processing and in contexts with restricted computational capabilities. VGG19's complexity presents issues, including the risk of overfitting, where the model excels on training data but struggles with unknown data. Model interpretability has become increasingly complicated, posing challenges in comprehending and elucidating the model's fundamental mechanisms and decision-making processes. These variables are essential for implementing and utilizing VGG-19 in different image recognition assignments.

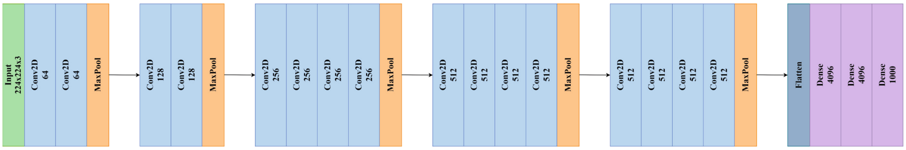

Fig. 2. VGG19: Enhanced VGG version with 19 layers for deeper feature analysis.

2.3 ResNet50

ResNet50 is a significant improvement in CNN structures, illustrated in Fig. 3 as a network with 50 layers that include shortcut connections. Residual connections, enhance the network's capacity to comprehend and analyze images. They accomplish this by facilitating faster and more precise feature detection, a critical element in several computer vision assignments.

Shortcut connections in ResNet50 are included to address the issue of the vanishing gradient problem commonly found in deep networks. ResNet50 prevents the degradation problem by enabling gradients to flow through shortcuts, ensuring that more depth leads to enhanced network performance without the saturation or diminishing feature reuse problems seen in conventional deep networks.

ResNet50 has around 23.6 million parameters, making it a good mix between computing performance and model complexity. This equilibrium allows for a versatile model that is useful for a range of applications without requiring the considerable resources usually needed for more complex structures.

Nevertheless, ResNet50 has limitations despite its gains. Overfitting may occur when the model is trained on little data. This implies that the model may excel in replicating the training data accurately but struggle to successfully apply its knowledge to new, unfamiliar data. Deploying ResNet50 on edge devices with restricted processing power and memory is an additional hurdle. The model's dimensions and computing demands may hinder its suitability for applications necessitating real-time processing on such devices. These factors are crucial when assessing the possible applications and implementation settings for ResNet50.ResNet50 significantly improves CNN structures, illustrated in Fig. 3 as a network with 50 layers that include shortcut connections. Residual connections enhance the network's capacity to comprehend and analyze images. They accomplish this by facilitating faster and more precise feature detection, a critical element in several computer vision assignments.

Shortcut connections in ResNet50 are included to address the issue of the vanishing gradient problem commonly found in deep networks. ResNet50 prevents the degradation problem by enabling gradients to flow through shortcuts, ensuring that more depth leads to enhanced network performance without the saturation or diminishing feature reuse problems seen in conventional deep networks.

ResNet50 has around 23.6 million parameters, making it a good mix between computing performance and model complexity. This equilibrium allows for a versatile model that is useful for a range of applications without requiring the considerable resources usually needed for more complex structures.

Nevertheless, ResNet50 has limitations despite its gains. Overfitting may occur when the model is trained on little data. This implies that the model may excel in replicating the training data accurately but struggle to successfully apply its knowledge to new, unfamiliar data. Deploying ResNet50 on edge devices with restricted processing power and memory is an additional hurdle. The model's dimensions and computing demands may hinder its suitability for applications necessitating real-time processing on such devices. These factors are crucial when assessing the possible applications and implementation settings for ResNet50.

2.4 InceptionV3

InceptionV3, a key member of Google's Inception series, is represented in Fig. 4. This model is distinguished by its use of inception modules, which are integral to its approach

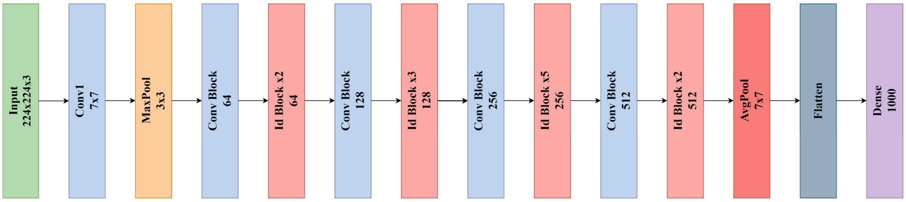

Fig. 3. ResNet50: 50-layer deep network with innovative skip connections.

to image analysis. These modules enable the network to handle and process image data more efficiently and effectively. Inception modules are designed to operate at various scales simultaneously, allowing the model to capture a wide range of features from different parts of an image.

One of the hallmark features of InceptionV3 is its ability to perform convolutions of different sizes concurrently within the same layer. This innovative approach means the network can capture both local and more extensive image context in a single pass, making it highly effective in recognizing patterns and objects in images, regardless of their scale.

The model's efficiency is further enhanced by batch normalization and factorization. Batch normalization helps stabilize and speed up the learning process, while factorization reduces the computational cost by decomposing larger convolutions into smaller, more manageable operations.

InceptionV3 comprises approximately 21.8 million parameters, a count that underscores its computational efficiency, especially when compared to other deep learning models with similar capabilities. This efficiency makes InceptionV3 a suitable choice for environments where computational resources are a concern. However, the model's streamlined complexity can sometimes be a limiting factor, particularly when dealing with highly complex patterns or diverse datasets. While it excels in general image recognition tasks, the architecture may require further adaptations or enhancements to maintain high performance in more specialized or challenging scenarios.

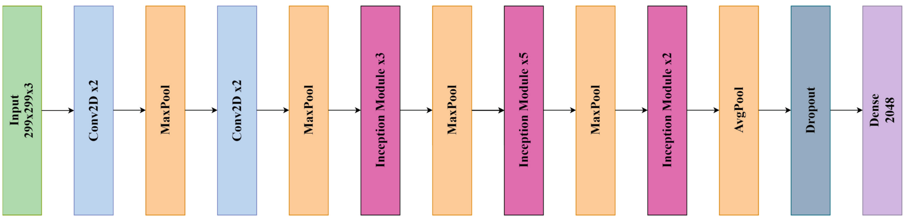

Fig. 4. InceptionV3: Efficient, parallelized feature extraction in image processing.

3 Methodology

In this research, we have developed a Convolutional Neural Network (CNN) model to categorize different types of content in cartoons. The model is adept at differentiating between regular scenes and those depicting violence or adultery. Our approach to creating this model was comprehensive and methodical, encompassing careful data preparation, rigorous model training, and thorough validation procedures. The entire process, from data acquisition to the final validation, is systematically outlined in our system architecture diagram, providing a clear visualization of the steps involved in developing this specialized CNN model. This structured methodology ensures that the model is accurate in its content categorization and robust and reliable for practical applications in content moderation.

3.1 System Architecture

The design of our Cartoon Filtering System is outlined in a complete workflow shown in Fig. 5, starting with a thorough phase of Data Collection. During this phase, we gather a wide range of data from different cartoon genres and media, including still photographs and moving video content. This type requires an adaptable method for preprocessing.

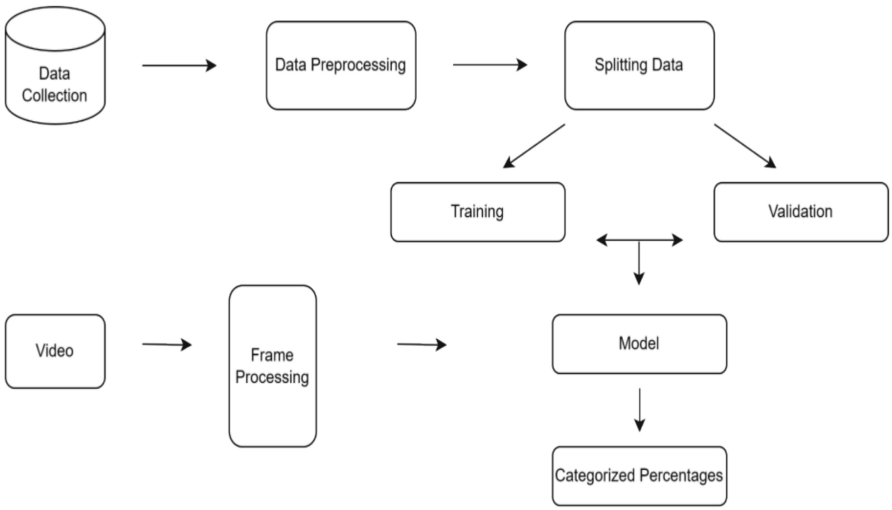

Fig. 5. Cartoon Filtering System Architecture - From data collection to content categorization.

During Data Preprocessing, various techniques, including image scaling and normalization, are used to adjust pixel values. We also utilize data augmentation techniques such as rotation and flipping to enhance the variety of the dataset. These procedures are essential for creating a model that is accurate and robust in handling the varied characteristics of cartoon images.

After preprocessing, we split the data by partitioning the dataset into separate training and validation sets. It is crucial to divide the data to objectively evaluate the model's performance, guaranteeing that the validation set appropriately represents the model's capacity to apply to fresh, unseen data.

Frame processing is a critical stage in video content production, where each video is broken down into separate frames. The frames are analyzed individually and as a series to maintain the context and continuity necessary for comprehending the narrative and nuances of each scene.

During the Training phase, iterative backpropagation and gradient descent techniques improve the CNN's weights. This phase focuses on refining the model to accurately recognize and understand intricate patterns and characteristics that distinguish distinct forms of information.

The validation phase simultaneously evaluates the model's predicted accuracy on the validation dataset. Consistent review and adjustment are crucial for refining the model and avoiding overfitting, ensuring its judgments are strong and relevant in practical situations.

The core of our technology is the model, a sophisticated network capable of identifying the nuances included in cartoon content. The model's structure incorporates advanced convolutional, pooling, and fully connected layers to capture the hierarchical properties specific to cartoon images.

This process concludes with the computation of Categorized Percentages, providing a numerical evaluation of the content's makeup. The percentages are crucial for analyzing the frequency of normal, violent, and adulterous scenes in cartoons, which helps in developing content filtering tactics and recommendations.

3.2 Proposed CNN Model Architecture

The proposed CNN model architecture combines VGG16's strong feature extraction skills with the computational economy of Inception models. The fusion process, shown in Fig. 6, starts with convolutional layers based on VGG16, known for accurately capturing fine details in images. The layers establish the groundwork for precise feature extraction at a high level of detail. The architecture includes modules similar to Inception, which improve the network's capacity to simultaneously evaluate information at different scales. This two-pronged method enables a comprehensive comprehension of visual material, encompassing both intricate details and broader contextual information.

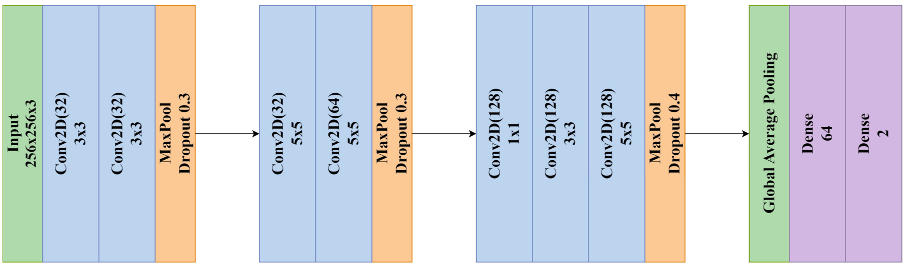

Fig. 6. Hybrid CNN Architecture - Integrating VGG16 and Inception models.

The initial segment of the network employs stacked convolutional layers, each with small receptive fields, following the structural blueprint of VGG16. The network increases its depth and improves its ability to identify intricate picture elements by adding convolutional layers before each max-pooling step without significantly raising the processing requirements.

The model advances from VGG16-inspired layers to Inception-style modules. The modules allow the network to analyze photos from several perspectives, capturing various attributes at the same time. This design characteristic is essential for recognizing both specific local features and broader patterns in the images.

The seamless incorporation of VGG16 and Inception modules achieves a balanced combination of network depth and breadth, enhancing overall performance. The architecture ends with global average pooling layers, which are succeeded by dense layers that combine and analyze the characteristics retrieved by earlier layers, ultimately leading to the final decision-making stage for content classification.

This architecture is designed to identify inappropriate content in cartoons for children by utilizing the depth of VGG16 and the breadth of Inception. The outcome is an extensive range of characteristics that may effectively categorize different types of information, therefore enhancing the safety of media consumption for youngsters.

3.3 Model Training and Validation

During the Training phase, our CNN model, equipped with a series of convolutional and pooling layers, engages with the preprocessed data to learn and identify complex patterns linked to different types of content. This phase employs a combination of forward propagation, where the model makes predictions on the training data, and backpropagation, a method used to adjust the model's parameters in response to the error rate. The model undergoes numerous epochs of training, each carefully calibrated to reduce the loss function and incrementally enhance the accuracy of content classification.

Simultaneously, the Validation phase plays a pivotal role in our methodology. This phase utilizes a distinct dataset not exposed to the model during training. This phase thoroughly evaluates the model's ability to generalize its learning to new, unseen data. By validating the model against this separate dataset, we can fine-tune its hyperparameters effectively, ensuring the integrity of the evaluation is maintained. This process is crucial to ascertain that the model can reliably and accurately differentiate between normal, violent, and adulterous content in various cartoon contexts.

3.4 Performance Metrics

Our CNN model's performance is thoroughly evaluated through two essential quantitative metrics: Accuracy and the F1-Score. Accuracy is a fundamental metric that indicates the overall rate at which the model correctly classifies content, reflecting its general effectiveness across all predictions. On the other hand, the F1-Score provides a more nuanced view by balancing precision (the model's ability to identify positive instances correctly) and recall (the model's capacity to detect all relevant cases). This balance is particularly valuable in situations where the distribution of content categories is uneven, ensuring that the model's performance is not biased towards any specific category.

Together, these metrics offer a comprehensive evaluation of the model's capabilities. Accuracy gives a straightforward measure of the model's effectiveness, while the F1-Score adds depth to the assessment, considering both the precision and recall aspects of the model's performance. Combining these metrics ensures a well-rounded evaluation, highlighting the model's proficiency in accurately categorizing content and its reliability in distinguishing between normal, violent, and adulterous scenes in cartoons. This multifaceted approach to performance evaluation is crucial in validating the model's applicability and effectiveness in real-world content moderation scenarios.

4 Implementation

The implementation of our CNN model encompassed a thorough and detailed process characterized by careful handling of data, utilization of a powerful computational infrastructure, and meticulous fine-tuning of the model to optimize its performance for the specific task of cartoon content classification. This section elaborates on the practical aspects of the model's application, beginning with the initial data collection and processing stages. It further delves into the setup of the computational environment, which played a crucial role in the model's training and efficiency.

Key emphasis is placed on the strategies adopted for optimizing the model, ensuring that it learns effectively from the training data and generalizes well to new, unseen content. This involves balancing achieving high accuracy and avoiding overfitting, a common challenge in machine learning models.

4.1 Dataset Collection and Preprocessing

Our training for the CNN model was supported by a dataset meticulously compiled from a wide array of anime series, aiming to ensure extensive coverage and representation of content types. This dataset consisted of 13,100 images, with a balanced classification of 6,800 'Child Friendly' and 6,300 'Non-Friendly'. These images were selected manually, spanning a broad spectrum of anime genres and styles to capture the rich diversity and thematic elements prevalent in animated media.

Dataset Integrity and Quality Assurance
Each image in the collection underwent meticulous cropping and scaling to achieve a standardized resolution of 256x256 pixels, ensuring consistency and quality. This resolution was carefully chosen to maintain crucial details necessary for accurate content classification while optimizing for computational efficiency, thereby facilitating effective learning from clear and consistent image data.

Preprocessing Techniques and Feature Extraction
In the preprocessing phase, we utilized the VGG16 preprocessing library, specifically designed to optimally prepare images for models built upon the VGG16 architecture. This involved adjusting the pixel values of images to a standardized range, enhancing the compatibility with our hybrid CNN model's input specifications, and improving the model's feature extraction process's efficacy.

To enhance the diversity of our training dataset and bolster the model's ability to generalize across various visual scenarios, we employed Keras's ImageDataGenerator. This tool played a pivotal role in applying real-time data augmentation techniques such as random rotations and horizontal flips, enriching the dataset's variability. Moreover, ImageDataGenerator streamlined the process of efficiently feeding batches of images to the model, markedly improving the training process's efficiency and effectiveness.

Dataset Collection Transparency and Features Utilized
The dataset's assembly was characterized by a transparent and methodical approach, laying a foundation of varied, high-quality data essential for the model's successful categorization of anime content into 'Child Friendly' and 'Non-Friendly.' For training the model, we extracted features, including visual attributes such as color, texture, and shape, through the application of a hybrid CNN architecture that leverages the strengths of InceptionV3 and VGG16 models. This enabled the nuanced detection and classification of complex visual patterns indicative of content suitability for children.

Through this detailed approach to dataset collection and preprocessing, we developed a robust model capable of accurately classifying anime content, underlining our commitment to creating safer viewing experiences for children.

4.2 Computational Environment

The training and development of our CNN models were executed on a high-performance computing system anchored by an AMD Ryzen 5800X CPU. This processor is renowned for its rapid execution capabilities, which are particularly vital for the computationally demanding process of training deep neural networks. The system's efficiency is further enhanced by its 64GB of RAM, which proves crucial in managing large datasets and complex model architectures, facilitating smooth data processing and manipulation.

Our computational setup operates on a Linux-based system, a platform chosen for its stability and adaptability, traits essential in research and development contexts. Linux's open-source nature allows for significant customization and optimization opportunities, making it a favored choice in scientific computing environments.

A pivotal component of our computational infrastructure is the NVIDIA RTX 3090 GPU, equipped with 24GB of VRAM. This GPU is notable for its 10,496 CUDA cores, which facilitate parallel processing, drastically accelerating the model training process. The substantial VRAM capacity enables the concurrent training of larger batches and more sophisticated models, ensuring efficient processing of our extensive dataset.

The harmonious integration of the Ryzen CPU, substantial RAM, and the high-performance RTX 3090 GPU forms an optimized ecosystem for deep learning tasks. This powerful combination guarantees that our models are trained swiftly and efficiently, providing us the agility to iterate and refine our models rapidly. This setup is integral to developing our deep learning models, ensuring they meet the high standards required for effective and accurate cartoon content classification.

4.3 Model Tuning and Optimization

Our CNN model underwent rigorous tuning and optimization to ensure peak performance. Key considerations in this process included selecting the optimal batch size and determining the appropriate number of training epochs. A batch size of 64 was chosen to strike a balance between computational efficiency and the network's performance capabilities. Over 60 epochs of training, the model was able to progressively adjust its weights and reduce the loss function, indicating improved learning and accuracy over time.

Batch Normalization and L2 Regularization

The implementation of batch normalization played a pivotal role in our optimization strategy. By normalizing the inputs for each layer, we mitigated internal covariate shifts, thus accelerating the training process and enhancing model stability and efficiency. L2 regularization was incorporated to tackle potential overfitting issues, adding a penalty to the loss function proportional to the square of the weights' magnitude. This encouraged smaller weight values, reducing model complexity and promoting generalizability.

Dynamic Learning Rate

We also adopted a dynamic learning rate approach, gradually reducing the learning rate as training progressed. This adaptive strategy allowed for finer weight adjustments as the model neared convergence, preventing overfitting to the training data.

Division of Dataset and Importance of Performance Metrics

The dataset was divided into a 75:25 ratio between training and validation sets, essential for comprehensive training while setting aside a substantial portion of data for validation. During the training phase, we closely monitored both accuracy and the F1 score to gauge the model's performance comprehensively.

Performance Metrics Explained

Accuracy. This metric serves as a direct measure of the model's overall performance, indicating the percentage of correct predictions out of all predictions made. Accuracy is particularly useful for providing a quick snapshot of model effectiveness. However, it may not fully capture the model's performance nuances in scenarios with class imbalances.

F1 Score. The F1 score is a more nuanced metric that balances precision (the proportion of true positive results in all positive predictions) and recall (the proportion of true positive results in all actual positives). This metric is crucial in scenarios where class imbalances might render accuracy less informative. By considering both false positives and false negatives, the F1 score provides a more balanced view of the model's predictive capability, making it particularly relevant for evaluating our CNN model's performance in distinguishing between 'Child Friendly' and 'Non-Friendly' content.

In summary, our approach to model tuning and optimization was designed to harmonize learning complexity with the capacity to generalize, ensuring the model's effectiveness and reliability in real-world applications. By utilizing both accuracy and the F1 score, we gained a holistic understanding of the model's performance, enabling us to fine-tune it for optimal real-world applicability.

5 Results and Analysis

This section marks the culmination of our meticulous approach to developing the CNN model, highlighting our implementation's significant outcomes and performance analysis. It encapsulates the rigorous process from the initial conceptualization to the practical application of the model. Here, we detail the results achieved through our systematic training, optimization, and application of the CNN model, providing insights into its effectiveness and efficiency in categorizing cartoon content. This comprehensive overview demonstrates the model's capabilities in a controlled environment and its potential applicability and reliability in real-world scenarios, particularly in the context of content moderation and classification.

5.1 Comparative Performance Analysis

The performance of our custom CNN model (denoted as CNN*) was evaluated in comparison to several established architectures (Table 1): VGG16, VGG19, ResNet50, and InceptionV3. This comparative analysis focused on two primary metrics, accuracy and the F1 score, assessed during the training and validation phases. Additionally, each model's total number of parameters was considered to provide insight into the model's complexity.

Observations:

- VGG16 and VGG19 showed high training (Tr) accuracy and F1 scores but exhibited a significant drop in performance during validation (Val), indicating a possible overfitting tendency.
- ResNet50 demonstrated the smallest difference (Diff) between training and validation metrics, suggesting better generalization capabilities, though its overall performance was lower compared to the VGG models.
- InceptionV3 balanced performance and generalization moderately well, with a mid-range difference between training and validation results.

Despite having significantly fewer parameters (|P|), our CNN* achieved competitive accuracy and F1 scores, particularly in the validation phase. This indicates a more efficient use of parameters and a balanced generalization ability.

Table 1. Comparative Analysis of Model Performance Metrics

Model	Accuracy (%)			F1-Score (%)			|P|
	Tr	Val	Diff	Tr	Val	Diff	
VGG16	**79.73**	71.48	8.25	**79.75**	71.75	8.00	15763858
VGG19	78.82	70.52	8.30	78.84	70.80	8.04	21073554
ResNet50	67.96	66.67	1.29	67.97	66.71	1.26	23591810
InceptionV3	77.32	**71.87**	5.45	77.37	**71.92**	5.45	24953122
CNN*	75.52	70.90	**4.62**	75.49	70.97	**4.52**	**2953410**

The comparative analysis highlights that while our CNN* model has fewer parameters, making it less complex and potentially more efficient, it still maintains a competitive performance regarding accuracy and F1 scores. This efficiency is particularly noteworthy in the context of deploying models in real-world applications where computational resources and efficiency are key considerations.

5.2 Video Processing and Result

The evaluation of our CNN* model's performance on new anime episodes involved a structured approach to classify content as 'Child-Friendly' or 'Not-Friendly.' The process included:

1. Video Segmentation: Segmenting the video into individual frames for detailed analysis.
2. Frame Processing: Classifying each frame using the CNN* model into 'Child-Friendly' or 'Not-Friendly' content.
3. Category Counting: Tallying the number of frames within each content category throughout the analysis.
4. Percentage Calculation: Determining the content distribution by calculating the percentage of frames in each category, thus quantifying the extent of 'Child-Friendly' versus 'Not-Friendly' content.

The results in Table 2 compare the CNN* model's classifications with human evaluations across seven distinct anime episodes. The comparative analysis shows a close correlation between the CNN* model's output and human judgment in most episodes. Notably, in episodes such as A4 and A5, the model shows a conservative approach to identifying 'Not-Friendly' content. Despite some variances, the general concordance suggests the CNN* model's capability to replicate human evaluators' decisions closely.

Table 2. Comparative Content Ratings per Episode - CNN Model vs. Human Judgment

Anime	CNN* Evaluation (%)		Human Evaluation (%)	
	Child-Friendly	Not-Friendly	Child-Friendly	Not-Friendly
A1	49.46	50.53	51	49
A2	32.33	67.67	34	66
A3	69.98	30.10	74	26
A4	7.02	92.99	28	72
A5	89.75	10.25	99	1
A6	93.74	6.26	98	3
A7	58.29	41.70	60	40

These observations confirm the model's potential utility in automated content classification, illustrating its value in assisting with content moderation efforts to safeguard appropriate media consumption for children. The slight differences between the models and human assessments provide constructive insights for future enhancements, aiming to refine the model's accuracy in identifying 'Child-friendly' content.

6 Conclusion and Future Work

6.1 Conclusion

Our study presented the development and application of a CNN model, referred to as CNN*, designed to categorize anime content into 'Child-Friendly' and 'Not-Friendly' categories. The comparative analysis of the model's performance against human evaluation revealed a high degree of consistency in content classification, affirming the model's potential as a tool for assisting in the moderation of child-appropriate content. Despite a few discrepancies in the classifications, which are expected given the subjective nature of content assessment, the CNN* model's performance holds promise for practical application in digital content platforms.

6.2 Future Work

Moving forward, our research will focus on several areas to enhance the model's functionality and applicability:

- Model Refinement: To address the discrepancies between the CNN* model and human evaluations, we will explore deeper neural network architectures and advanced machine learning techniques. This may involve more sophisticated forms of transfer learning and the integration of contextual information to understand the content's narrative better.
- Dataset Expansion: We plan to augment our dataset with various anime genres and styles, including more nuanced content categories. This expansion will aim to improve the model's robustness and accuracy across a more comprehensive array of content.
- Real-time Processing: Efforts will be made to optimize the model for real-time content classification. This includes reducing computational requirements and improving processing speed without compromising accuracy.
- Interdisciplinary Collaboration: Working with child psychologists and media experts will provide insights into fine-tuning the classification criteria and ensuring that the model's content categorization aligns with psychological understandings of child-friendliness.
- Ethical Considerations: As part of our commitment to responsible AI development, we will examine the ethical implications of automated content classification and address potential biases in the model.
- User Feedback Integration: Incorporating user feedback mechanisms can provide real-world data on the model's performance and assist in iterative improvements.

By addressing these areas, we aim to advance the field of automated content moderation, creating safer online environments for children and enhancing the overall quality of digital content curation.

References

1. Khalil, T., Bangash, J.I., Khan, A.W., Lashari, S.A., Khan, A., Ramli, D.A.: Detection of violence in cartoon videos using visual features. Procedia Comput. Sci. **192**, 4962–4971 (2021). https://doi.org/10.1016/j.procs.2021.09.274
2. Khan, M., Tahir, M.A., Ahmed, Z.: Detection of violent content in cartoon videos using multimedia content detection techniques. In: 2018 IEEE 21st International Multitopic Conference, pp. 1–5 (2018)
3. Khan, N., Hussain, A., Khan, A., Uddin, I.: Categorized Violent Contents detection in cartoon movies using Deep Learning Model: mobile Net. Presented at the (2019)
4. Taha, M.M., Alsammak, A.K., Zaky, A.B.: InspectorNet: Transformer network for violence detection in animated cartoon. Eng. Res. J. **52**, 114–119 (2023). https://doi.org/10.21608/erjsh.2023.181713.1119
5. Honarjoo, N., Abdari, A., Mansouri, A.: Violence detection using pre-trained models (2021). https://doi.org/10.1109/IPRIA53572.2021.9483558
6. Sumon, S.A., Goni, R., Hashem, N. Bin, Shahria, T., Rahman, R.M.: Violence detection by pretrained modules with different deep learning approaches. Vietnam J. Comput. Sci. **07**, 19–40 (2019). https://doi.org/10.1142/S2196888820500013
7. Jain, A., Vishwakarma, D.: State-of-the-arts Violence Detection using ConvNets (2020). https://doi.org/10.1109/ICCSP48568.2020.9182433
8. Arun Akash, S.A., Sri Skandha Moorthy, R., Esha, K., Nathiya, N.: Human violence detection using deep learning techniques. J. Phys. Conf. Ser. **2318**, 12003 (2022). https://doi.org/10.1088/1742-6596/2318/1/012003
9. Bermejo, E., Deniz, O., Bueno, G., Sukthankar, R.: Violence detection in video using computer vision techniques. Comput. Anal. Images Patterns. 332–339 (2011)
10. Jahlan, H.M. Bin, Elrefaei, L.A.: Detecting Violence in Video Based on Deep Features Fusion Technique (2022)
11. Yousaf, K., Nawaz, T.: A deep learning-based approach for inappropriate content detection and classification of youtube videos. IEEE Access **10**, 1 (2022). https://doi.org/10.1109/ACCESS.2022.3147519
12. Zhou, Y., Jin, Y., Luo, A., Chan, S., Xiao, X., Yang, X.: ToonNet: a cartoon image dataset and a DNN-based semantic classification system. In: Proceedings of the 16th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry. Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3284398.3284403
13. Rani, P., Kaur, J., Kaswan, S.: Automatic video classification: a review. EAI Endorsed Trans. Creat. Technol. **7**, 163996 (2018). https://doi.org/10.4108/eai.13-7-2018.163996
14. Singh, S., Tyagi, B.: Computational comparison of CNN based methods for violence detection (2023). https://doi.org/10.21203/rs.3.rs-3130914/v1
15. Ding, C., Fan, S., Zhu, M., Weiguo, F., Jia, B.: Violence detection in video by using 3D convolutional neural networks (2014). https://doi.org/10.1007/978-3-319-14364-4_53
16. Abdelkader, M., Abbass, B.: Violence detection enhancement by involving convolutional block attention modules into various deep learning architectures : comprehensive case study for UBI-fights dataset. IEEE Access **11**, 37096–37107 (2023). https://doi.org/10.1109/ACCESS.2023.3267409
17. Sumon, S.A., Goni, R., Hashem, N. Bin, Shahria, T., Rahman, R.M.: Violence detection by pretrained modules with different deep learning approaches. Vietnam J. Comput. Sci. **07**, 19–40 (2020). https://doi.org/10.1142/S2196888820500013
18. Irfanullah, Hussain, T., Iqbal, A., Yang, B., Hussain, A.: Real time violence detection in surveillance videos using Convolutional Neural Networks. Multimed. Tools Appl. **81**, 38151–38173 (2022). https://doi.org/10.1007/s11042-022-13169-4