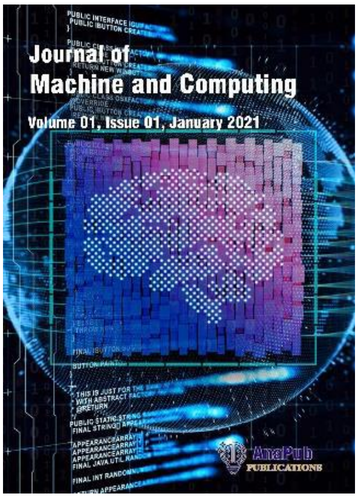
Journal Pre-proof



Impact of Synthetic Data on Training and Improved YOLOv8 Models for Helmet Detection

Mohd Arshad and Pradeep Kumar

DOI: 10.53759/7669/jmc202505114
Reference: JMC202505114
Journal: Journal of Machine and Computing.

Received 05 June 2024
Revised form 16 February 2025
Accepted 08 May 2025

Impact of Synthetic Data on Training and Improved YOLOv8 Models for Helmet Detection

[1]Mohd Arshad, [2]Pradeep Kumar
[1,2]Department of Computer Science and Information Technology,
Maulana Azad National Urdu University, Hyderabad, India
[1]azmi.arshad9044@gmail.com, [2]pradeep@manuu.edu.in

Abstract - Acquiring real-time, accurate, large datasets is crucial and time-consuming for specific problems. Numerous datasets are available with annotations, but most are not feasible for a special task because of differences in the class label, class imbalance, and variability. One such solution to this problem is to use artificially crafted datasets (or synthetic datasets), which are scalable and can be automatically annotated. We utilized two different approaches—stable diffusion and cut-paste-blend—to generate a synthetic dataset. This study investigates the use of synthetic image datasets to observe the performance of YOLOv8 and improved YOLOv8 models for helmet detection. We trained models on both real-world and synthetic datasets and evaluated their performance in terms of detection accuracy. After training 50 epochs, the model achieved an mAP@50 of 78.6% on real data, 45.5% on synthetic data, and 75.4% on hybrid datasets. We analyzed how the hybrid dataset affected results using different ratios and discovered that with a 3:1 mix of hybrid data, the YOLOv8-based model reached an mAP@50 of 90.3%, which is better than when real and synthetic data were used in equal amounts. We proposed the Convolutional Block Attention Module-based YOLOv8-CBAM to enhance the accuracy of helmet and non-helmet detection. Experimental results indicate that YOLOv8-CBAM achieved an mAP@50 of 91% at 50, which is 0.7% better than the baseline model. This study also indicates that the correct proportion of synthetic datasets solved the class imbalance problem and improved the helmet detection accuracy in challenging environments.

Keywords -Synthetic Data, YOLO, Attention Mechanism, Deep Learning, Helmet Detection

I. INTRODUCTION

In computer vision, data and training are the foundation of a model's success. Data and training together ensure the development of robust, accurate, and generalized systems. Object detection tasks need training datasets comprising images and annotations of bounding boxes to indicate the presence of objects inside an image. Manual annotation is extensively employed for this purpose; however, it is a labor-intensive procedure.

Publicly available real-world datasets have some challenges: availability, variety, privacy, and compatibility with the task. Classes with different labels or conditions may be underrepresented, leading to imbalanced training data that fails to capture specific edge cases that negatively impact model performance. For developing an object detection model, a high-resolution image with a class label is required that covers various scenarios, such as variations in lighting conditions, occlusions, and environmental factors. Artificially generated or synthetic data, which mimics real-world scenes, is one such solution. The motive behind using synthetic image datasets typically relates to addressing challenges or gaps in real-world data for specific applications.

There is a pressing need for synthetic image datasets in applications like detecting traffic rule violations, where the availability of annotated images is limited. Synthetic datasets provide the flexibility to generate large-scale data, simulate a wide range of environmental conditions, and augment underrepresented classes, ensuring better model training and generalization. Current research indicates that the use of synthetic data, especially in healthcare [1], agricultural[2], transportation, and autonomous vehicles [3], has increased exponentially. It can be created through various techniques, such as data augmentation, simulations, or generative AI [4].

The amount of data for a specific problem is crucial. Fewer datasets can cause underfitting, while overly large datasets may lead to overfitting; both negatively impact the model's performance. Even balanced-size datasets may also suffer due to the absence of diverse scene conditions, such as variations in viewpoints, backgrounds, and environmental factors, and lead to hindering the model's generalization capabilities. Object detection algorithms require datasets in specific formats to ensure compatibility with the training frameworks and the algorithms' data processing pipeline. **Table 1** displays some of the popular data formats used for object detection algorithms and frameworks.

Table 1. Summary of Algorithms, Format and Annotation types

Algorithm	Preferred Format	Annotation Type
YOLO	Text Files (.txt)	Normalized bounding boxes
Faster R-CNN	Pascal VOC (.xml), COCO	Bounding boxes in XML or JSON
SSD	Pascal VOC, COCO	Bounding boxes in XML or JSON
RetinaNet	COCO	Bounding boxes in JSON

Detectron2	COCO	Bounding boxes in JSON
TensorFlow API	TFRecord	Serialized protobuf files
Open Images	CSV	Bounding boxes in CSV

In this study, a realistic synthetic dataset is aimed to be created for object detection, particularly helmets and non-helmets. A proposed methodology was formed to generate synthetic data using two different approaches, named Stable Diffusion and Cut-Paste-Blend, to generate and detect helmets or non-helmets in the intelligent transport system for detecting traffic rule violations. The contribution of this paper is two-fold and aims to i) propose a method to artificially generate object instances that represent various scenes for helmet detection and ii) model performance analysis on the hybrid dataset (real and synthetic) with the state-of-the-art deep learning algorithms. The study also aims to focus on how well synthetic datasets generalize the task using YOLOv8 and improved YOLOv8 and their effects on detection accuracy while integrating different types of attention mechanisms in the base architecture.

II. LITERATURE REVIEW

In this segment of our study we reviewed previous publications to investigate current research trends, gaps and shortcomings of earlier works. It introduces the need and significance of using synthetic data for this study. Our work aims to analyse how synthetic datasets effect the performance of model compare to real datasets. We have done literature review on recent studies that are discussed in details in this section. We analyzed the publicly accessible datasets included in **Table 2** to present an overview of datasets and to demonstrate the necessity for synthetic datasets.

Synthetic data generation can be done in two ways either statistical methods or deep learning methods. Synthetic images are generated by statistical methods by modeling real data distribution and generating samples that share the same visual pattern. Gaussian Mixture Models (GMMs) and Markov Random Fields(MRFs) are two popular techniques for generating synthetic images using a statistical approach. Deep Learning methods help to create image and text based datasets using different approaches such as Generative Adversarial Networks (GANs), Diffusion Models, Denoising Diffusion Probabilistic Models (DDPM), Neural Style Transfer, Variational Autoencoders (VAEs) and Large Language Models (LLMs). A comprehensive analysis of various synthetic data generation technologies has been done and found that every method has some strength, challenges and advantages depending on the type of data. This study suggested that the different methods and technologies can be used for image data generation but these are computationally intensive and challenging to train [4]. Authors Kniazev et al.[5], proposed an object detection model using synthetic image datasets. They generated realistic synthetic images with 3D models, 3D camera images, background images, noise images and animation effects. The study also shows that performance of the detection model is highly dependent on the ratio of synthetic to real data.

In a related study, Ljungqvist M. G. et al. [6] used Centered Kernel Alignment (CKA) to look at similarity and the effects of training on synthetic data layer by layer. The results revealed that the early stage of training yielded the most similarity, while the frozen layer showed almost no difference. In a different study, Kim J. et al. [7] used hybrid datasets to test how well the model worked on synthetic data by recognizing scaffolding objects in images. This study achieved an object recognition accuracy of 88.6%. Wang, Y., et al. [8] developed a synthetic datasets comprising both photo-realistic and non-photo realistic images to create an object detector using Faster R-CNN. The efficacy of the object detector on synthetic data is inferior to that on a real datasets. They employed transfer learning to boost the detector's performance on synthetic data. The study[9] explores different ways to generate synthetic data for training of traffic sign detector, including random placement of different quality signs.

Object detection algorithm YOLO divides the images into grids. It emerged in 2016 with many updates, the most recent version labeled as YOLOv11. Another YOLO family - YOLOv8 has noteworthy components like mosaic data augmentation, anchor free detection, C2f module, decoupled head and modified loss function. Network architecture, anchor free detection, training strategies, and the decoupled head approach together make YOLOv8 a state-of-the art deep learning model for object detection. Previous work [10–13] has shown that attention mechanisms can marginally improve the accuracy of the model in the YOLO design. By improving performance approximately 9.34% over YOLOv3, TA-YOLO [10] is based on YOLOv3. On real datasets, improved - YOLOv5 [11], YOLOv5 with squeeze and excitation block [12], and YOLOv8n-SLIM-CA [13] showed model's performance improved by 3%, 2.5%, and 3.54%, respectively.

Table 2 Summary of some most popular Real and Synthetic Helmet datasets for traffic surveillance

Ref	Datasets	Type	No of Image	Resolution	Env. condition	Annotation?	CI?	Remark
[14]	HelmetML	R	28736	768 × 576	diverse	No	No	4 different types of helmet data captured: half, full, modular and off-

								road
[15]	Helmet Detection	R	764	mixed	Day time	Yes	No	It has 2 classes: with helmet & without helmets that contains helmet & head instances
[16]	SynPeDS	S	NA	1920x1280	NA	Yes	No	It contains pedestrian datasets for traffic analysis.
[17]	SHEL5K	R	5000	416 x 416	diverse	Yes	Yes	Improved version of SHD - has 6 class lables.
[18]	Hardhat Dataset	R	7063	NA	NA	Yes	Yes	Three class labels: helmet, head, and person. There are no proper labels on the person class in the given datasets.

*CI - class imbalance, **R- Real, ***S-Synthetic

Publicly available datasets, such as those cited in [20-21], frequently exhibit class imbalance problems that might hinder the accuracy of deep learning models. Moreover, gathering data customized for a particular problem, such as helmet detection or face detection, is a complicated and time-consuming task. Privacy issues intensify the challenges of data collection in these areas. We can utilize synthetic data to train models and enhance generalization to overcome these issues. There are a few synthetic datasets; however, they mostly concentrate on traffic scenarios and do not specifically annotate helmet factors. Some of the key challenges we have identified through literature surveys are -

- Need for synthetic helmet datasets that better fit the algorithm and give better generalizations.
- Robust methods and techniques are required to create synthetic datasets.
- Examine the impact of hybrid data(ratio of synthetic & real) and evaluate how it affects the model's performance (YOLOv8 vs YOLOv8 with attention mechanism).

III. METHODOLOGY

Traditional methods like cut-paste and cut-paste-blend are used to produce synthetic data for image segmentation tasks. Some advanced deep learning-based tools, such as DALL-E 3, generate realistic visuals based on textual descriptions. Another text-to-image generation method is the Stable Diffusion model used for generating artificial data. In the proposed methodology, as shown in **Figure 1,** we used both Stable Diffusion and the cut-paste-blend method to generate synthetic datasets for the object detection task. Next, we divide the datasets into subsets and develop various YOLO-based models. Synthetic data is used to train the models on rare or difficult-to-observe events. The final stage analyzes the performance of the model and the effects of synthetic data.

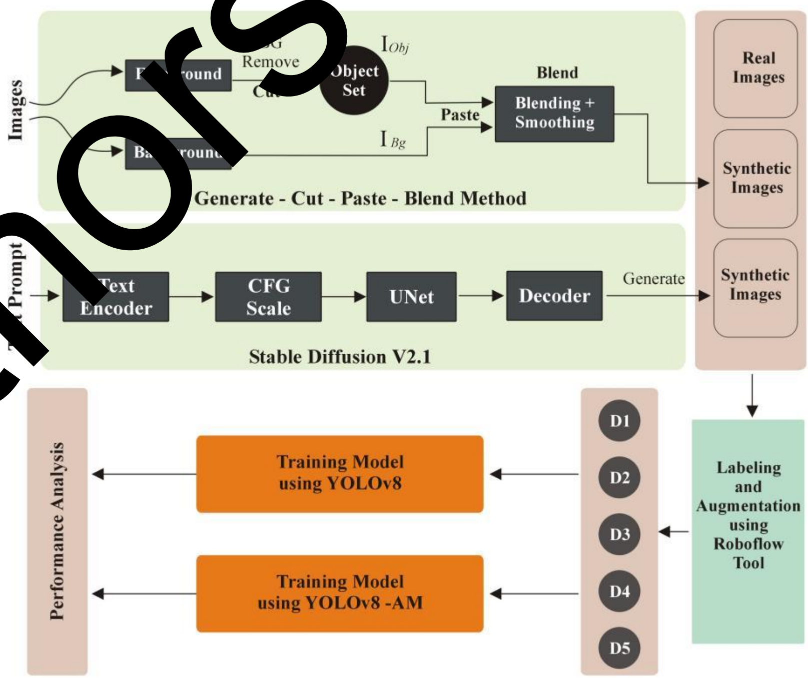


Figure 1 Proposed methodology

The proposed method uses Stable Diffusion and Cut-Paste-Blend to generate synthetic datasets. First, we were provided with a prompt, a negative prompt, and CFG (Classifier-Free-Guidance) scale hyperparameters to control the model and ensure it strongly follows the given prompt. We have defined different variables: riders:{male, female}, environments:{urban, rural, traffic, dense, sparse}, lightening_conditions:{day, mid-day, night}, vehicle_type:{motorcycle, scooter, bicycle}, and helmet:{any type of helmet, rider without helmet}. We randomly selected a value from the parameters and subsequently generated synthetic data using the Stable Diffusion model. Another technique we utilized is the cut-paste-blend method, where cut" means extracting desired objects (I_{obj}) from images, paste" means pasting object instances to the background (I_{bg}), and blend" means blending object instances with the background. For object extraction, we removed the background and converted it into PNG format using the OpenCV library. Pasting objects directly onto the background can cause pixel artifacts, leading to an unnatural appearance. In the blending phase, we used alpha blending with a Gaussian blur for soft edges and seamlessly integrated the background and foreground to achieve a natural appearance. We also tried some advanced blending methods, such as Poisson image blending, which blends the texture and gradient of a background image into the background. Various transformations are applied, including random rotation between -30 and +30, random scaling between 1.0x and 1.5x, random positioning on the background, and blending to generate variability and realism in the synthetic datasets for model training.

Synthetic Data Generation

The process of collecting datasets consists of two main stages: data acquisition and annotation. The first stage of dataset collection includes acquiring images from available platforms, including public repositories and databases, alongside internet resources. It remains a difficult task to find datasets with particular object instances. Manually collected datasets are necessary to train robust machine learning models, although producing them requires an immense amount of time, energy, and resources and demands attention to various object types, especially in safety-critical fields like traffic surveillance. Synthetic data enables scalable solutions to domain-specific problems, like traffic rule violations—helmet detection—when real-world annotated data is difficult to obtain due to privacy laws, logistical limitations, or prohibitive costs. Researchers have pursued numerous experiments to evaluate the effectiveness of training models using a combination of real and synthetic data. Several synthetic datasets, Synthia, VKITTI, and GTAV, exist for different tasks, but to the best of our knowledge, there is no synthetic dataset that handles traffic rule violations (i.e., helmet detection or triple riding detection). Diffusion models are generative models that help us create fake or unrealistic images. Some of the key diffusion models are DDPM (Denoising Diffusion Probabilistic Models), LDM (Latent Diffusion Models), Stable Diffusion, Imagen (Google's Model), and DALLE-2 (OpenAI's Model). In this study, we have collected 480 real images from Kaggle and 276 images taken from MANUU. The diffusion models [19]-[21] and cut-paste-blend method are used to generate 756 synthetic datasets. **Figures 2(a), 2(b),** and 2**(c)** represent synthetic images and real images.

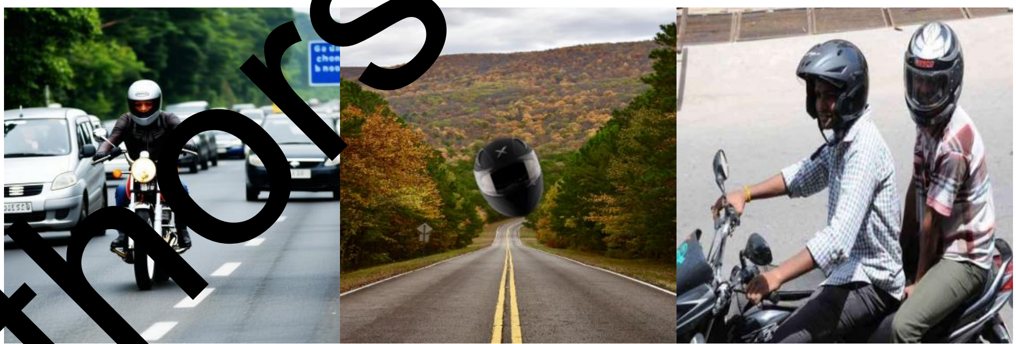


Figure 2(a) Synthetic image (Stable Diffusion), 2**(b)** Synthetic image (Cut-Paste-Blend),2**(c)** Real Image

The division of datasets into different proportions enables researchers to measure how dataset size and ratio impact model accuracy as well as overall performance outcomes. Table 3 splits the datasets into different ratios to help train various YOLO-based models and evaluate their performance systematically. The method enables the identification of optimal dataset ratios that produce maximum practical outcomes.

Table 3. Description of real and synthetic datasets with varying proportions

Datasets	Synthetic Images	Real Images
D1	756	0
D2	504	252
D3	378	378
D4	252	504
D5	0	756

Improved YOLOv8

YOLOv8 is a state-of-the-art object detection deep learning model. Further, we can improve the accuracy of the detection model by modifying the architecture of YOLOv8. In our study, we incorporated an attention mechanism in the base architecture and analyzed the performance differences. **Figure 3** shows the basic architecture of YOLOv8.

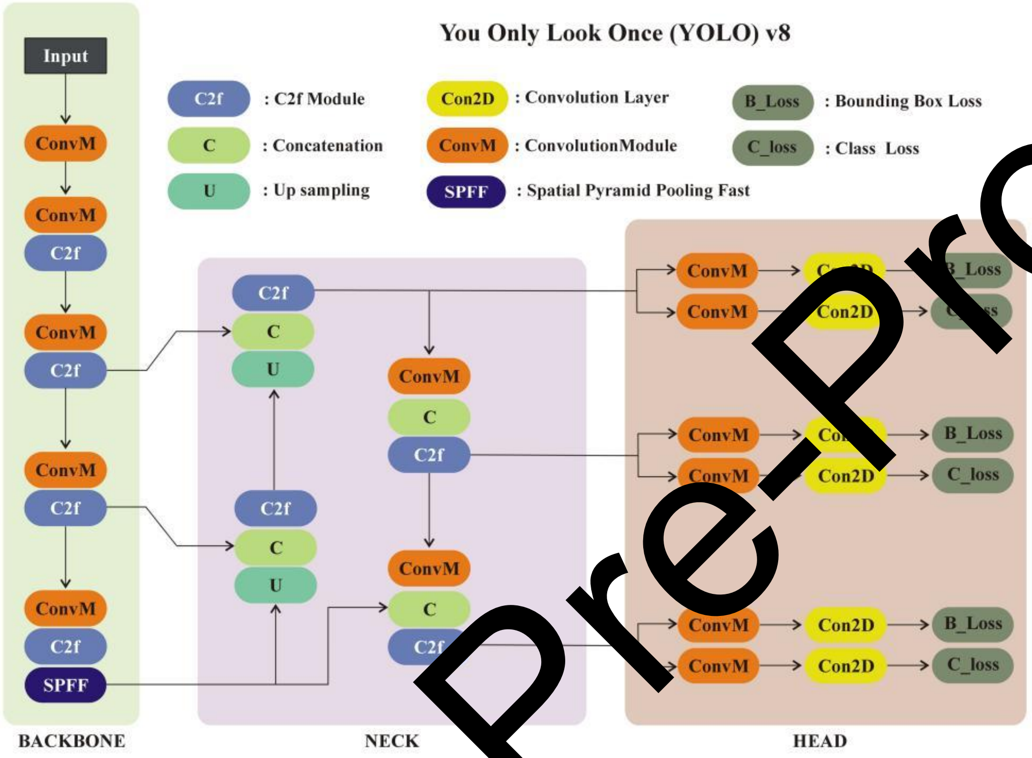


Figure 3 Architecture of YOLOv8

The attention mechanism attempts to mimic human behavior by focusing on more important information while disregarding irrelevant details. Practically, it forces networks to concentrate only on important parts. Integration of attention mechanisms into the YOLOv8 architecture sometime improves the object detection accuracy by focusing on essential features while suppressing irrelevant ones [22].

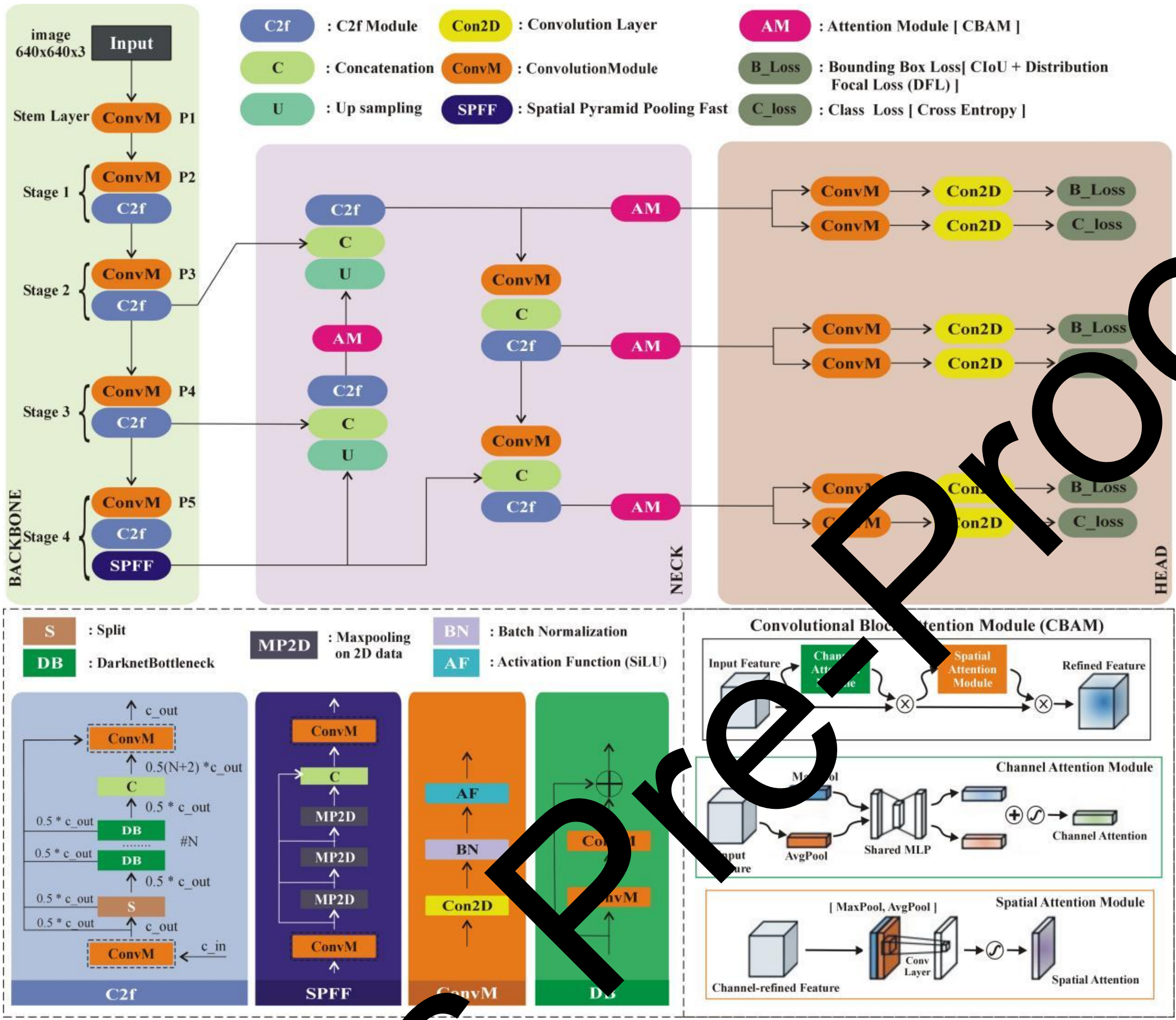


Figure 4 Detailed illustration of improved YOLOv8 AM model, where Attention Modules (AM): CBAM is utilized. This diagram is inspired by the research work presented in [22][23].

The proposed model retains YOLOv8's backbone (C2f blocks, SPPF) and integrates Convolutional Block Attention Modules (CBAM) at critical feature aggregation points in the head. The neck employs a bidirectional feature pyramid network, where up-sampled features are concatenated with backbone outputs and processed by C2f blocks, followed by CBAM to sequentially refine channel and spatial attention. Final detections are generated via three heads attached to CBAM. **Figure 4** represents the detailed architecture of the improved version of YOLOv8 with the CBAM attention mechanism. In this study, we substitute CBAM in the base architecture and analyze the performance changes on real data and synthetic data.

IV. RESULT AND DISCUSSION

All calculations were performed, and training of the YOLOv8-based helmet detection model was done in Google Colab's cloud services with an especially useful Nvidia T4 GPU with 16 GB VRAM and TPUv2 for faster processing. We formed a hybrid dataset from real and synthetic images to make the models robust to diverse traffic scenarios. We trained the models on 640 x 640 RGB images in batches of 32, with a learning rate of 0.01. We used a patience value of 25 during model training to prevent overfitting. This setup demonstrated the effectiveness of cloud-based training and the value of real and synthetic data in addressing traffic rule violations like helmet usage detection.

Table 4 Impact of Epochs on Baseline vs. Transfer Learning (tf) Models

YOLOv8 Model	Epochs 'n'	mAP50		
		Helemt	Non Helmet	All
Baseline	30	0.834	0.722	0.778

	40	0.801	0.735	0.768
	50	**0.850**	**0.721**	**0.786**
	60	0.843	0.712	0.776
	70	0.831	0.698	0.765
Transfer Learning	30	0.845	0.676	0.758
	40	0.832	0.698	0.765
	50	**0.836**	**0.705**	**0.770**
	60	0.829	0.690	0.723
	70	0.825	0.677	0.751

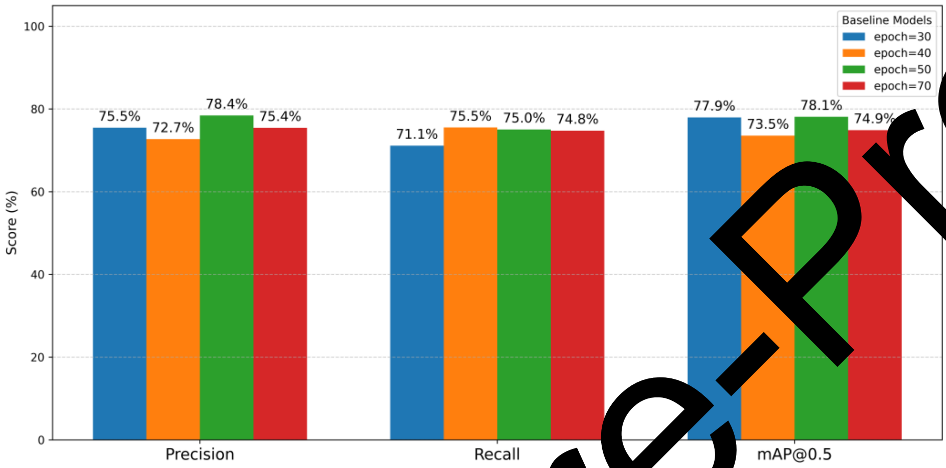


Figure 5 Impact of Epoch Count on Precision, Recall and mAP@0.5 in YOLOv8 Training

To evaluate performance variation, we trained the YOLOv8 model with and without transfer learning. In the multi-stage training approach, the model's backbone was frozen by setting freeze=10, which prevents changes to the first 10 layers that correspond to the feature extraction component of the architecture. This stage focused on training only the detection head over 15 epochs, allowing the model to learn task-specific features without altering the pre-trained backbone parameters. In the subsequent stage, the best-performing weights obtained from the first stage were loaded and the model was fine-tuned end-to-end by unfreezing all layers (i.e., setting freeze=0). This procedure allowed the model to jointly optimize the backbone and detection head, enhancing feature representation and improving detection accuracy. **Table 4** shows the trade-off between epochs and accuracy.

The best overall mAP@0.5 of baseline and transfer learning models occurs at 50 epochs. The baseline model achieved 0.786 (helmet: 0.850) and transfer learning achieved 0.770 (helmet: 0.836). Baseline performs slightly better at early epochs and is better than transfer learning at higher epochs. **Figure 5** represents the impact of epochs on various performance metrics.

Object detection results on real, synthetic and hybrid datasets

Initially, we took only real datasets, and the datasets were split into training and testing. We trained the model on 70% of the datasets and used the remaining datasets for testing. Similarly, we developed another model by considering all synthetic datasets. Later, we took 50% real and 50% synthetic data to form a hybrid dataset and developed another model. **Table 5** displays the effectiveness of the model. We applied different evaluation metrics to assess the quality of the model's outcomes.

Table 5 Results of object detection model using real, synthetic and hybrid data

Datasets	**Labels**	**Precision**	**Recall**	**mAP0.5**	**mAP:0.95**	**F1 Score**
Real	Helmet	0.812	0.845	0.850	0.554	0.828
	Non Helemt	0.635	0.762	0.721	0.417	0.693
	All	**0.723**	**0.804**	**0.786**	**0.486**	**0.761**
Synthetic	Helmet	0.769	0.597	0.707	0.473	0.672
	Non Helemt	0.266	0.313	0.203	0.108	0.287
	All	**0.518**	**0.455**	**0.455**	**0.290**	**0.484**
Hybrid	Helmet	0.815	0.866	0.897	0.626	0.839
	Non Helemt	0.568	0.687	0.611	0.351	0.621
	All	**0.691**	**0.776**	**0.754**	**0.488**	**0.731**

The least effective performance was obtained by training detection methods using only synthetic data, which indicates the need to use a mixed or hybrid dataset approach. To verify, either a mixed dataset may improve the detection performance or not. A model is trained using a hybrid dataset; the helmet class label improved the detection accuracy. However, the overall accuracy of the model was highest when it was trained on real-world data. Use of a hybrid dataset is particularly helpful for leveraging the model to learn more specific concepts, improve generalization, and deal with class imbalance. **Table 5** illustrates the effect on the performance of the trained model using real, synthetic, and hybrid datasets.

Estimation of Synthetic Data Amount

The synthetic data amount for training the YOLOv8s and modified YOLOv8-CBAM architecture was determined through an ablation study comparing model performance across varying ratios of real-to-synthetic data.

Table 6 Ratio of synthetic and real images in the performance of helmet detection

Datasets	mAP50(%)		
	Helmet	Non Helemt	All
D1 [S:4, R:0]	70.7	20.3	45.5
D2 [S:3, R:1]	89.2	60.8	74.9
D3 [S:2, R:2]	89.7	61.1	75.4
D4 [S:1, R:3]	**90.3**	**69.6**	**79.9**
D5 [S:0, R:4]	85.0	72.1	78.6

S- Synthetic data, R- Real data

Table 6 illustrates the contribution of the proportion of synthetic and real images to the performance of the helmet detection model. Overall mAP@50 on Dataset D1, which only consists of synthetic images, was only 45.5%, as it was the worst performing in the Non-Helmet class (20.3%). Real images significantly improved performance. D2 (3:1 synthetic-to-real ratio) achieved an overall mAP@50 of 74.9%, while D3 (2:2 ratio) achieved an mAP@50 of 75.4%. D3 also enhanced the balance of class-wise consistency. These results highlight the limitations of training with synthetic data alone and the necessity of using real data variations for robust detection. The dataset D4, which used three times more real data than synthetic data, achieved an overall mAP@50 of 79.9% and performed better in detecting helmets (90.3%) and non-helmets (69.6%). Interestingly, the overall mAP@50 for D5, which only had real images, was also 79.9%. The result implies that a combination of synthetic and real data can be used not only to improve generalization and learning of complex concepts. It also helps to better handle the class imbalance problem. Our experiments show that using a mix of real and synthetic data for helmet detection works better when there are more real images and fewer synthetic ones.

YOLOv8 vs YOLOv8-AM: Results on Hybrid Data

We explored the effect of attention mechanisms on the performance of object detection using the YOLOv8 architecture. Two different training strategies were employed on these cases: (a) training from scratch and (b) fine-tuning a pre-trained model. First, a baseline YOLOv8 model with no attention mechanism is trained from scratch. Then, the Convolutional Block Attention Module (CBAM) was incorporated into the model to evaluate how it would affect it. To improve contour detection at the end, C2f_DySankeConv (Dynamic Snake Convolution) was combined with the backbone, while CBAM was used in the neck to further improve the representation and the detection accuracy. To evaluate the consistency and effectiveness of the attention mechanism in the transfer learning setting, the same sequence of model configurations was also applied to the fine-tuned pretrained YOLOv8 model. **Table 7** shows the comparative analysis of YOLOv8 and YOLOv8-AM using hybrid data.

Table 7 Comparative Analysis of YOLOv8 Variants Using Hybrid Data

Model	**Attention Mechanism**	**Learnig**	**Layers**	**Parameters (in Million)**	**GFlops**	**Inference (ms)**	**mAP50 (%)**		
							Helmet	**Non Helemt**	**All**
YOLOv8	**N/A**	**S**	168	3.00	8.1	7.6	80.3	49.3	64.8
	CBAM	**S**	201	3.01	8.1	2.8	85.1	46.1	65.6
	DSC+CBAM	**S**	233	3.53	8.4	5.4	84.0	45.8	64.9
YOLOv8s	**N/A**	**TF**	168	11.12	28.4	5.3	90.3	69.6	79.9
	CBAM	**TF**	**201**	**11.14**	**28.5**	**7.3**	**91.0**	**70.6**	**80.8**
	DSC+CBAM	**TF**	233	12.01	28.9	7.1	90.7	69.9	80.0

This study shows a comparative analysis of various YOLOv8-based variants, which proves that the YOLOv8-CBAM model outperforms other YOLOv8-based models. We achieve more accurate object detection on custom datasets by integrating the Convolutional Block Attention Module (CBAM) for feature refinement. We use both training schemes: the training from scratch and the fine-tuning. The table showed the comparison of the performance of YOLOv8 and its attention-incorporating variants under two learning strategies: training from scratch (S) and transfer learning (TF). The YOLOv8-CBAM model is better than the baseline and other variants in both training setups. By training from scratch, YOLOv8-CBAM attains the highest helmet detection accuracy (85.1%) rather than the baseline YOLOv8 (80.3%) and has the highest overall mAP@50 (65.6%). Although YOLOv8-DSC+CBAM performs the best among scratch-trained models on helmet detection accuracy (84.0%), its overall performance is slightly worse because its non-helmet detection is only 45.8%. All models significantly exceed their performance under the transfer learning setup. As with YOLOv8-CBAM, the overall mAP was again led by the YOLOv8-CBAM model with an overall score of 80.8% and conditioned helmet (91.0%) and non-helmet (70.6%) detection scores. While the YOLOv8-DSC+CBAM variant also performs competitively but with an mAP@50 of 80.0% overall, it fails to outperform the CBAM-only version. Furthermore, adding CBAM and DSC to the model increases its complexity (from 11.14M parameters to 12.01M parameters and 28.9 GFLOPs) without significantly affecting inference time, which takes about 7 ms. Our results indicate that the CBAM integration makes the most consistent and effective improvement on YOLOv8-based models, especially if we use transfer learning, which points out the benefit of attention mechanisms in getting excellent object detection performance from models trained on a hybrid set.

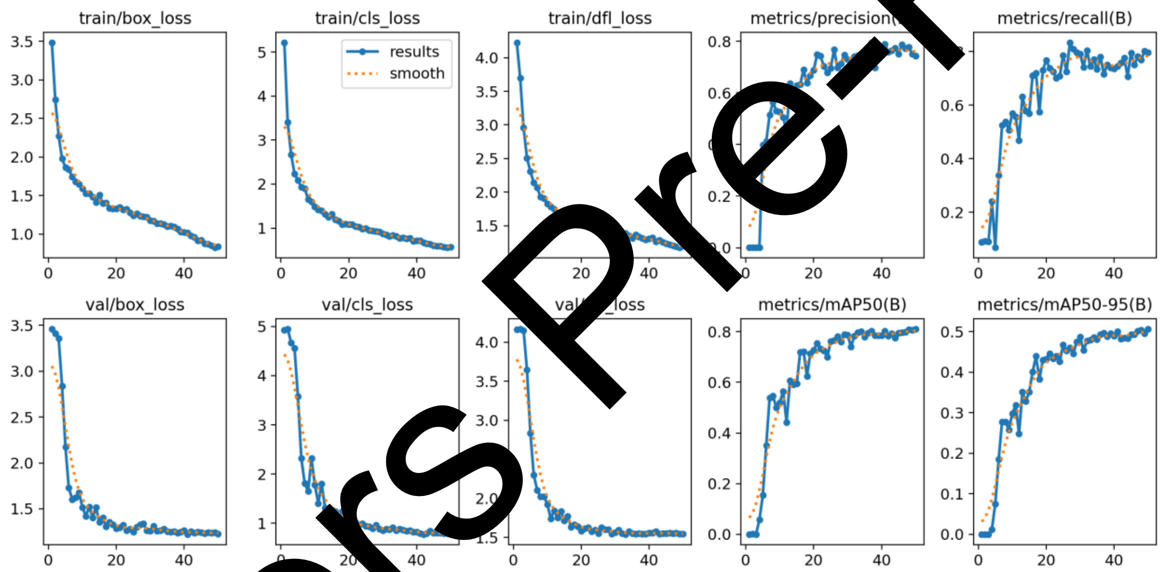


Figure 6 Loss profile and evaluation metrics of the YOLOv8-CBAM on the hybrid dataset

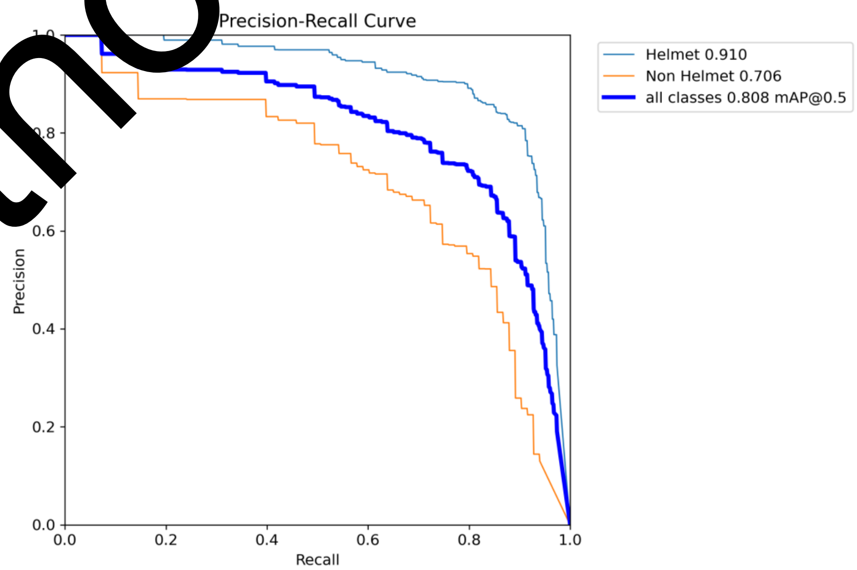


Figure 7 The precision-recall curve of the YOLOv8-CBAM

Figure 8 Helmet and Non-helmet detection results

The effectiveness of the developed model is demonstrated through various performance metrics depicted in the accompanying figures. Two types of losses and key evaluation metrics are presented in **Figure 6.** Box loss is the difference between the predicted and actual bounding boxes, and classification loss is the success rate of predicting whether an object is wearing a helmet or not. The decreasing trends of both losses indicate that the network has been trained successfully and has learned the required features well. All four evaluation metrics get improved and, asymptotically, along with training, finally reach high values. This result shows that the model not only learns well but also performs well in terms of generalization, which implies robustness and reliability of the model in real-world situations.

To further emphasize the improvements provided by the developed model, **Figure 7** shows the PR curves. A widely used metric for classifying performance evaluation is the area under the PR curve (PR AUC), which is a larger value indicating a better trade-off between precision and recall. We demonstrated that the proposed work achieves the largest PR AUC among all compared models, confirming its superiority in classification. **Figure 8** displays the detection outcomes of YOLOv8-CBAM. It is observed that the developed model detects even small or partially visible targets (helmet or non-helmet) in the detection outputs. The result demonstrates that the proposed approach is highly effective and reliable in real-world tasks, and its ability to accurately classify such targets in complex scenarios greatly supports the adequacy of the model.

Table 8 Helmet detection performance comparison of various studies

Model	**Recall**	**mAP@50**	**Params/M**	**GFlops**
YOLOv5- BEH [14]	83.1	89.2	14.47	56.8
YOLOv8-SLIM-CA [13]	83.2	88.5	2.79	11.4
FGP-YOLOv8 [24]	83.8	89.6	2.41	6.6
Ours (YOLOv8s-CBAM)	**82.3**	**91.0**	**11.14**	**28.5**

The performance comparison in Table 8 shows that the detection accuracy of YOLOv8-CBAM is higher than other models. YOLOv5-BEH [14] showed high recall but lower mAP@50 than our proposed model. Also, computationally heavy, making it inefficient for real-time application. YOLOv8-SLIM-CA [13] is lightweight and efficient, but its performance is slightly lower than other models. FGP-YOLOv8 [24] is more efficient and lighter than other models and is useful for edge devices due to minimal computational cost. Our model achieves 91.0% mAP@50, outperforming all other models in detection precision. CBAM improves feature selection, making it more robust in complex scenes despite a minor recall trade-off.

V. CONCLUSION & FUTURE WORK

In this study, we created a small and diverse synthetic dataset for helmet detection in the intelligent transport system (ITS) domain. We conducted an analysis on synthetic datasets using YOLOv8 and YOLOv8-AM, which can be utilized to develop a robust system for detecting traffic rule violations. We addressed some of the gaps—collecting traffic scene data and helmet-specific data to generate synthetic datasets tailored to helmet detection tasks. Training with a 1:3 synthetic-to-real data ratio (D4) yielded the best results, proving that real-world data is essential for robust generalization, while synthetic data helps mitigate class imbalance. This study indicates that YOLOv8-CBAM is better than other models at detecting helmets, reaching the highest mAP@50 score of 91.0% while keeping computing costs reasonable. Using synthetic data helps address privacy concerns and enhances data availability for model training. Synthetic image datasets are a useful way to improve helmet detection with YOLO models, and future studies could look into using this method for other traffic rule violations like detecting triple riding, wrong-way driving, and fancy number plates to create a combined model that identifies all these violations.

CRediT Author Statement

The authors confirm contribution to the paper as follows:
Conceptualization: Arshad M, Kumar P; **Methodology:** Arshad M, Kumar P; **Software:** Arshad M; **Data Curation:** Arshad M; **Writing-Original Draft Preparation:** Arshad M, Kumar P; **Visualization:** Arshad M; **Investigation:** Kumar P; **Supervision:** Kumar P; **Validation:** Arshad M, Kumar P; **Writing- Reviewing and Editing:** All authors reviewed the results and approved the final version of the manuscript.

Data Availability The Datasets used and /or analysed during the current study available from the corresponding author on reasonable request.

Conflicts of Interests
The author(s) declare(s) that they have no conflicts of interest.

Funding
No funding agency is associated with this research.

Competing Interests
There are no competing interests

References

[1] Giuffrè, M., & Shung, D. L. (2023). Harnessing the power of synthetic data in healthcare: innovation, application, and privacy. *Npj Digital Medicine*, *6*(1). https://doi.org/10.1038/s41746-023-00927-3

[2] Giakoumoglou, N., Pechlivani, E. M., & Tzovaras, D. (2023). Generate-Paste-Blend-Detect: Synthetic dataset for object detection in the agriculture domain. *Smart Agricultural Technology*, *5*, 100258. https://doi.org/10.1016/j.atech.2023.100258

[3] Delgado, G., Cortés, A., García, S., Loyo, E., Berasategi, M., & Aranjuelo, N. (2023). Methodology for generating synthetic labeled datasets for visual container inspection. *Transportation Research Part E Logistics and Transportation Review*, *175*, 103174. https://doi.org/10.1016/j.tre.2023.103174

[4] Goyal, M., & Mahmoud, Q. H. (2024). A Systematic review of Synthetic data generation techniques using Generative AI. *Electronics*, *13*(17), 3509. https://doi.org/10.3390/electronics13173509

[5] Kniazev, A., Slivnitsin, P., Mylnikov, L., Schlechtweg, S., Perm National Research Polytechnic University, & Anhalt University of Applied Sciences. (2021). Influence of synthetic image datasets on the result of neural networks for object detection. *Proc. Of the 9th International Conference on Applied Innovations in IT*, 55. https://icaiit.org/proceedings/9th_ICAIIT_1/2_2_Kniazev.pdf

[6] Ljungqvist, M. G., Nordander, O., Skans, M., Mildner, A., Liu, T., & Nugues, P. (2023). Object Detector Differences when Using Synthetic and Real Training Data. *SN Computer Science*, *4*(3). https://doi.org/10.1007/s42979-023-01704-5

[7] Kim, J., Wang, I., & Yu, J. (2024). Experimental study on using synthetic images as a portion of training dataset for object recognition in construction site. *Buildings*, *14*(5), 1454. https://doi.org/10.3390/buildings14051454

[8] Wang, Y., Deng, W., Liu, Z., & Wang, J. (2019). Deep learning-based vehicle detection with synthetic image data. *IET Intelligent Transport Systems*, *13*(7), 1097–1105. https://doi.org/10.1049/iet-its.2018.5365

[9] Shakhuro, V., Faizov, B., & Konushin, A. (2019). Rare traffic sign recognition using synthetic training data. In *In Proceedings of the 3rd International Conference on Video and Image*

Processing (ICVIP '19). Association for Computing Machinery, New York, NY, USA, 23–26. https://doi.org/10.1145/3376067.3376105

[10] Zhou, Q., Qin, J., Xiang, X., Tan, Y., & Xiong, N. N. (2021). Algorithm of helmet wearing detection based on AT-YOLO Deep Mode. *Computers, Materials & Continua/Computers, Materials & Continua (Print)*, *69*(1), 159–174. https://doi.org/10.32604/cmc.2021.017480

[11] Shan, C., Liu, H., & Yu, Y. (2023). Research on improved algorithm for helmet detection based on YOLOv5. *Scientific Reports*, *13*(1). https://doi.org/10.1038/s41598-023-45383-x

[12] Xu, Z. P., Zhang, Y., Cheng, J., & Ge, G. (2022). Safety helmet wearing detection based on YOLOV5 of attention mechanism. *Journal of Physics Conference Series*, *2213*(1), 012038. https://doi.org/10.1088/1742-6596/2213/1/012038

[13] Lin, B. (2024). Safety helmet detection based on improved YOLOV8. *IEEE Access*, *12*, 28260–282[illegible]2. https://doi.org/10.1109/access.2024.3368161

[14] Patil, K., Jadhav, R., Suryawanshi, Y., Chumchu, P., Khare, G., & Shinde, T. (2024). HelmetM[illegible]: A dataset of helmet images for machine learning applications. *Data in Brief*, *5[illegible]*, 1107[illegible]. https://doi.org/10.1016/j.dib.2024.110790

[15] Helmet detection. (2020). [Dataset]. In *Helmet detection*. Kaggle. https://www.kaggle.com/datasets/andrewmvd/helmet-detection

[16] T. Thamaraimanalan, P. G. Vishnu, R. Dineshkumar, A. Dayanand, S. [illegible] Shahil and N. [illegible]shokkumar, "Prevention of Road Accidents Using Hybrid Machine Learning Algorithm," 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 2137–2143, Mar. 2024, doi: 10.1109/icaccs60874.2024.10717245.

[17] Otgonbold, M., Gochoo, M., Alnajjar, F., Ali, L., Tan, T., Hsieh, J., & Chen, [illegible] (2022). SHEL5K: an extended dataset and benchmarking for safety helmet detection. *Sensors*, *22*(6), 2315. https://doi.org/10.3390/s22062315

[18] Xie, L. (2019). Hardhat [Dataset]. In *Harvard Dataverse*. https://doi.org/10.7910/dvn/7cbgos

[19] Reutov, I. (2023). Generating of synthetic datasets using diffusion models for solving computer vision tasks in urban applications. *Procedia Computer Science*, *229*, 335–344. https://doi.org/10.1016/j.procs.2023.12.036

[20] Pezoulas, V. C., Zaridis, D. I., Mylona, [illegible], Androutsos, [illegible] Apostolidis, K., Tachos, N. S., & Fotiadis, D. I. (2024). Synthetic data generation methods in healthcare: A review on open-source tools and methods. *Computational and Structural Biotechnology Journal*, *23*, 2892–2910. https://doi.org/10.1016/j.csbj.2024.07.005

[21] Corvi, R., Cozzolino, D., Zingarini, G., Poggi, G., Nagano, K., & Verdoliva, L. (2023). On the detection of synthetic images generated by diffusion models. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. https://doi.org/10.1109/icassp[illegible].10095167

[22] Chien, C., Ju, R., Chou, K., Xieerke, [illegible], & Chiang, J. (2025). YOLOV8-AM: YOLOV8 based on effective attention mechanisms for pediatric wrist fracture detection. *IEEE Access*, 1. https://doi.org/10.1109/access.2025.3549839

[23] Woo, S., Park, [illegible], Lee, [illegible] & Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. In *Lecture notes in computer science* (pp. 3–19). https://doi.org/10.1007/978-3-030-01234-2_1

[24] Zhang, [illegible], Ma, [illegible], Huang, J., Zhang, C., & Gao, X. (2025). An improved lightweight safety helmet detection algorithm for YOLOV8. *Computers, Materials & Continua/Computers, Materials & Continua (Print)*, *[illegible]0*(0), 1–10. https://doi.org/10.32604/cmc.2025.061519