International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

HIGH-PRECISION AERIAL OBJECT DETECTION MODEL UTILIZING YOLO V10 DEEP NEURAL NETWORK

Ayman Javid Hussain[1], Dr. Lalitha Saroja CH[2], Ruqiya Fatima[3]
[1]PG Scholar, Department of AI&DS, Shadan Women's College of Engineering and Technology, Hyderabad,
aymanjhussain@gmail.com
[2]Assoc Professor, Department of CSE, Shadan Women's College of Engineering and Technology
lalithasarojathota@gmail.com
[3]Asst Professor, Department of CSE, Shadan Women's College of Engineering and Technology
ruqiyafatima2716@gmail.com

ABSTRACT
The installation of a real-time visual tracking system with an active pan-tilt camera for indoor human motion detection is presented in this study. For precise and effective human detection in every video frame, the system makes use of the cutting-edge YOLOv10 object detection model. YOLOv10 is perfect for real-time applications due to its fast inference and enhanced detection accuracy, especially in difficult illumination and occlusion situations. The system uses a multiple object tracking (MOT) framework that keeps a dynamic graph structure to guarantee robustness. Several theories about the quantity and temporal trajectories of identified persons are handled by this graph. YOLOv10 allows frame-wise object detection with fewer false positives and missing detections than traditional frame differencing techniques. In order to achieve consistent tracking throughout time, the MOT module performs temporal data association, checking and confirming YOLOv10's frame-wise predictions. Because of this close interaction, the tracker can estimate object placements and increase overall tracking reliability by giving feedback to the detection module. In order to choose the most plausible explanation for the observed video, tracking hypotheses are continuously expanded and trimmed. The system's efficacy in real-time human motion tracking scenarios is demonstrated by experimental results, which show that the inclusion of YOLOv10 significantly improves detection precision and temporal consistency.

1. INTRODUCTION
Object detection is one of the most important and rapidly evolving research areas in the field of computer vision and artificial intelligence. It involves identifying and localizing objects of interest within images or video streams. With the rapid advancement of deep learning technologies, object detection systems have become increasingly accurate, efficient, and capable of operating in real-time environments. Among various application domains, aerial image analysis has gained significant attention due to its critical role in surveillance, traffic monitoring, environmental observation, disaster management, military reconnaissance, agriculture, and smart city development.

Aerial images captured by satellites, drones, and unmanned aerial vehicles (UAVs) provide a comprehensive view of large geographical areas. However, detecting objects within aerial imagery presents several challenges. Objects often appear at different scales, orientations, and resolutions due to varying altitudes and camera perspectives. Additionally, factors such as background clutter, occlusion, illumination changes, weather conditions, and dense object distributions make accurate object detection a complex task. Traditional computer vision techniques based on handcrafted features struggle to handle these challenges effectively, leading to limited detection performance in real-world applications.

Recent advancements in deep learning have significantly improved object detection capabilities. Convolutional Neural Networks (CNNs) have emerged as the foundation of modern object detection systems due to their ability to automatically learn meaningful feature representations from large-scale datasets. Among the various deep learning-based object detection frameworks, the YOLO (You Only Look Once) family has become one of the most popular approaches because of its ability to achieve high detection accuracy while maintaining real-time processing speed. YOLO models perform object localization and classification within a single network architecture, enabling efficient end-to-end object detection.

YOLOv10 represents the latest advancement in the YOLO series and introduces several architectural improvements designed to enhance both accuracy and computational efficiency. Unlike previous versions, YOLOv10 adopts an anchor-free detection mechanism, reducing computational complexity and improving localization precision. The model incorporates enhanced feature extraction capabilities, optimized multi-scale feature fusion, and a decoupled detection head that independently handles classification and localization

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

tasks. These improvements enable YOLOv10 to achieve superior detection performance while maintaining high inference speed, making it particularly suitable for aerial image analysis applications.

The primary objective of this project is to develop a fine-grained object detection system for aerial images using the YOLOv10 deep neural network architecture. The proposed system aims to accurately detect and classify objects of varying sizes and categories while operating efficiently in real-time environments. To improve robustness and generalization, advanced preprocessing techniques, data augmentation methods, and transfer learning strategies are incorporated during the training process. The system is designed to handle challenging aerial scenarios involving small objects, complex backgrounds, and varying environmental conditions.

The proposed framework leverages the strengths of YOLOv10 to achieve reliable object detection performance across diverse aerial datasets. By combining efficient feature extraction, multi-scale learning, and optimized training strategies, the system provides accurate localization and classification results while maintaining computational efficiency. The developed model has potential applications in aerial surveillance, intelligent transportation systems, environmental monitoring, precision agriculture, disaster response operations, and autonomous aerial platforms.

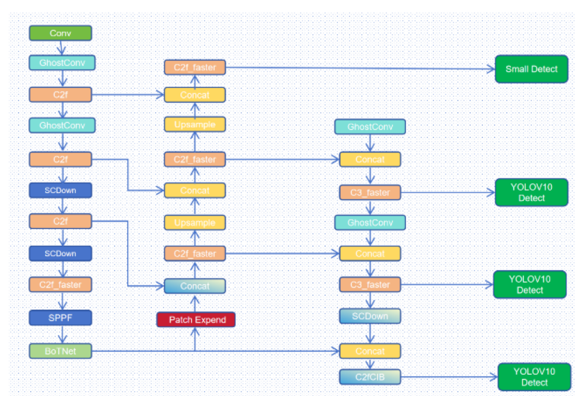
The remainder of this work presents the system architecture, methodology, model training procedures, experimental evaluation, performance analysis, and future enhancements of the proposed YOLOv10-based aerial object detection framework. The results demonstrate the effectiveness of modern deep learning techniques in addressing the challenges associated with fine-grained object detection in aerial imagery and highlight the potential of YOLOv10 for next-generation intelligent vision systems.

2. EXISTING SYSTEM OF PROJECT

Multiple object tracking bas been a challenging research topic in computer vision. It has to deal with the difficulties existing in single object tracking, such as changing appearances, non-rigid motion; dynamic illumination and occlusion, as well as the problems related to multiple object tracking including inter-object occlusion, multi-object concision. There has been much work on multiple object visual tracking. they use a sampling algorithm for tracking fixed number of objects. Tao et al. [2] present an efficient hierarchical algorithm to track multiple people.

ALGORITHM (YOLOv5):

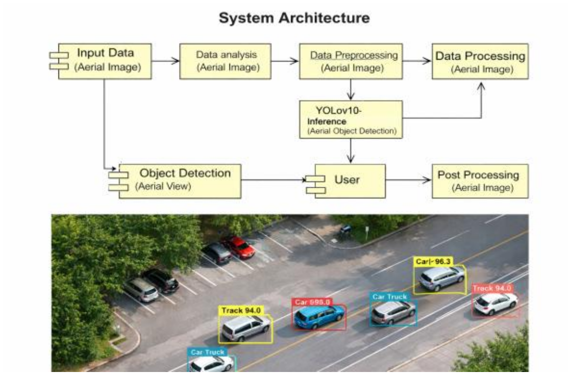
The existing system utilizes YOLOv5 (You Only Look Once version 5), a widely adopted and high-performance real-time object detection model developed by Ultralytics. YOLOv5 builds upon the success of earlier YOLO versions, offering a balance between speed and accuracy through a highly optimized PyTorch-based implementation. It features a modular and scalable architecture with multiple model sizes—YOLOv5n (nano), YOLOv5s(small), YOLOv5m (medium), YOLOv5l (large), and YOLOv5x (extra-large)—to accommodate various hardware and performance requirements. The system leverages an anchor-based detection mechanism, a CSPDarknet53-inspired backbone for feature extraction, a PANet for feature aggregation, and a decoupled head for object classification and bounding box regression. With pre-trained weights available for common datasets like COCO, YOLOv5 achieves high mean Average Precision (mAP) while maintaining low inference latency, making it suitable for real-time applications such as surveillance, autonomous vehicles, industrial automation, and medical diagnostics. The model also includes powerful training utilities, built-in data augmentation techniques like mosaic and mixup, and compatibility with export formats like ONNX, CoreML, and TensorRT for easy deployment across diverse platforms. However, despite its strong performance, YOLOv5 requires manual tuning of anchor boxes and relies on NMS (Non-Maximum Suppression) during post-processing, which can increase computational complexity in certain scenarios. Nevertheless, its robust accuracy, ease of use, and active community support make



SYSTEM ARCHITECTURE:

YOLOv5 a reliable foundation for object detection tasks in many existing systems.

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper



3. PROPOSED SYSTEM

YOLOv10, the latest advancement in the "You Only Look Once" object detection series, represents a significant leap forward in real-time object detection by combining high accuracy, exceptional speed, and deployment efficiency in a single unified architecture. Introduced in 2024, YOLOv10 was designed to overcome the limitations of earlier models by unifying the training and inference processes, thereby simplifying deployment and improving overall consistency in performance. Unlike previous versions that often-required separate post-processing or anchor-based mechanisms, YOLOv10 embraces an anchor-free detection approach and integrates a fully decoupled head for classification and localization tasks. It also incorporates an efficient backbone and improved feature aggregation modules that allow for better representation of objects at multiple scales, enhancing detection precision for small and large objects alike.

ALGORITHM (YOLOv10):

The model introduces optimized loss functions, label assignment techniques, and advanced data augmentation strategies such as mosaic and mixup to ensure robust learning and generalization. YOLOv10 comes in various sizes—Nano, Small, Medium, and Large—enabling developers to select the right balance of speed and accuracy based on deployment constraints, whether on edge devices like mobile phones and NVIDIA Jetson or powerful GPUs. It significantly improves upon previous versions in terms of mean Average Precision (mAP), frames per second (FPS), and computational efficiency, offering a superior performance-to-complexity ratio. With support for ONNX, TensorRT, TorchScript, and OpenCV integration, YOLOv10 is highly versatile and easy to deploy across various platforms. Its real-world applications span across diverse domains such as autonomous driving, surveillance, robotics, medical imaging, industrial automation, and smart city infrastructures. Ultimately, YOLOv10 stands as a powerful testament to how deep learning models can be both high-performing and efficient, making it a top choice for next-generation object detection systems.

4. DATASET PREPARATION:

In this study, the PASCAL VOC dataset and the MS COCO dataset are used for experimentation. The PASCAL VOC dataset is composed of PASCAL VOC2007 and PASCAL VOC2012, with 20 objec. categories in total. This dataset consists of 16 551 images for training and 4952 images for testing. The MS COCO 2017 dataset contains 80 object categories and is divided into training, validation, and test sets. The training set is made up of 118 287 images, the validation set consists of 5000 selected images, and the test set comprises 20 000 unlabeled images. For the PASCAL VOC dataset, the initial step is to convert it to the text format recognizable by YOLO. Next, the anchor box size for YOLOv5s is determined by making reference to the MS COCO dataset. The default anchor box size is no longer applicable for the PASCAL VOC dataset with objects of varying scales and aspect ratios. The k-means clustering algorithm is used to re-cluster the bounding boxes in the dataset for a 640 pixel × 640 pixel image. The final anchors obtained are shown in Table 4. The data augmentation methods used in this study include the Mixup[39] and Mosaic methods. The Mixup method increases the size of the training dataset by applying a linear transformation to the input image/data. The Mosaic data augmentation technique combines four images that have undergone Mixup operations in a random arrangement. These methods can reduce GPU memory usage, accelerate network training, enrich the dataset, increase the number of small objects to be sampled, and improve the overall robustness of the model industrial environments. noise

5.TECHNIQUES USED IN PROJECT

In order to identify irregularities in industrial machine sounds, this study combines machine learning, deep learning, feature extraction, and audio signal processing. Preprocessing the audio recordings is the initial stage in order to eliminate noise, adjust sound levels, and get the data ready for analysis. In order to extract features and train models, this guarantees that the machine sound signals are clear and reliable.

Mel-Spectrogram and Mel-Frequency Cepstral Coefficients (MFCCs) are two key feature extraction techniques used to accurately represent the audio data. The Mel-Spectrogram allows the system to record changes in frequency over time by converting sound signals into a time-frequency representation. MFCCs are

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

widely used in speech and sound recognition applications because they extract the most important spectral characteristics of the audio signal. These characteristics aid in the model's ability to differentiate between typical and unusual machine operating circumstances. The project utilizes several machine learning algorithms, including Support Vector Machine (SVM), Decision Tree (DT), and Extreme Gradient Boosting (XGBoost). These algorithms learn patterns from labeled sound data and classify machine sounds as either normal or anomalous. Among these models, XGBoost demonstrates strong performance due to its ensemble learning capability and ability to handle complex patterns in the data.

The research uses Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) networks for deep learning. These models are especially made to handle sequential data and record temporal relationships in audio signals. Because GRU requires less processing power than LSTM and efficiently learns long-term dependencies, it is chosen as the main model. The model is well suited for industrial sound anomaly detection because of its update and reset gate methods, which allow it to preserve crucial information while discarding unnecessary data. The entire system follows a supervised learning approach, where the models are trained using labeled datasets containing both normal and anomalous machine sounds. During training, the models learn to identify patterns associated with normal machine operation and detect deviations that may indicate faults or failures.

Lastly, common metrics like Accuracy, Precision, Recall, F1-Score, and Confusion Matrix are used to assess the models' performance. These assessment methods aid in determining how well the system detects unusual machine noises and reduces incorrect predictions.

In conclusion, the project builds an intelligent and dependable industrial sound anomaly detection system for predictive maintenance and fault diagnosis by integrating audio preprocessing, Mel-Spectrogram, MFCC, SVM, Decision Tree, XGBoost, RNN, LSTM, and GRU approaches.

6. MODEL TRAINING

The performance of the proposed aerial object detection system heavily depends on the effectiveness of the training process. In this work, the YOLOv10 deep neural network is trained on a large-scale annotated aerial image dataset containing multiple object categories captured from different altitudes, viewing angles, and environmental conditions. The objective of the training phase is to enable the model to accurately detect and classify objects of varying sizes while maintaining real-time inference speed.

Prior to training, all images undergo a comprehensive preprocessing stage. The collected aerial images are resized to a fixed input resolution compatible with the YOLOv10 architecture. Image normalization techniques are applied to standardize pixel values and improve convergence during optimization. To increase dataset diversity and enhance the model's robustness against overfitting, several data augmentation techniques are employed, including horizontal flipping, vertical flipping, random rotation, scaling, cropping, mosaic augmentation, color jittering, and brightness adjustments. These augmentation strategies enable the model to generalize effectively to unseen aerial scenes.

The dataset is divided into three subsets: training, validation, and testing. Approximately 70% of the dataset is allocated for training, 20% for validation, and the remaining 10% for testing. The training set is utilized for learning object representations, while the validation set is used to monitor model performance and prevent overfitting during training. The testing set is reserved exclusively for final performance evaluation.

YOLOv10 utilizes an anchor-free object detection framework that eliminates the need for predefined anchor boxes, thereby reducing computational complexity and improving localization accuracy. The network consists of an efficient backbone for feature extraction, a feature aggregation neck for multi-scale information fusion, and a decoupled detection head responsible for simultaneous object classification and bounding box regression. This architecture allows the model to detect both small and large objects effectively within aerial imagery.

Transfer learning is employed by initializing the network with pre-trained weights obtained from large-scale benchmark datasets. This approach significantly reduces training time and improves feature extraction capabilities. During training, the model learns hierarchical visual features through forward propagation, while the error between predicted outputs and ground-truth annotations is minimized using backpropagation.

The optimization process is performed using the Stochastic Gradient Descent (SGD) optimizer or Adaptive Moment Estimation (AdamW) optimizer. The learning rate is initialized to an optimal value and gradually adjusted using a learning rate scheduling mechanism to ensure stable convergence. Batch normalization and regularization techniques are

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

incorporated to improve training stability and prevent verfitting.

The YOLOv10 loss function consists of multiple components that collectively optimize detection performance. The classification loss measures the accuracy of object category predictions, the localization loss evaluates bounding box regression quality, and the confidence loss assesses object presence within predicted regions. The total loss function can be expressed as:

Total Loss = Classification Loss + Localization Loss + Confidence Loss

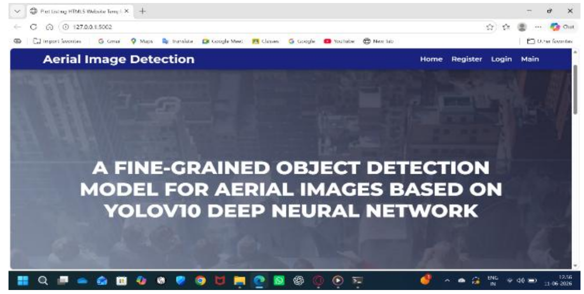
Throughout the training process, performance metrics such as Precision, Recall, F1-Score, Mean Average Precision (mAP), and Intersection over Union (IoU) are continuously monitored. Precision measures the proportion of correctly detected objects among all detections, while Recall evaluates the ability of the model to identify all relevant objects. Mean Average Precision serves as the primary evaluation metric for object detection performance.

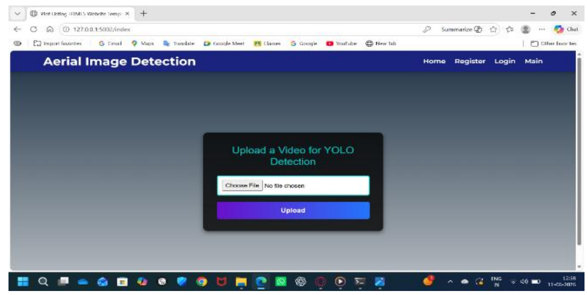
Training is conducted for multiple epochs until convergence is achieved. During each epoch, the model processes batches of images, updates network parameters, and evaluates performance on the validation dataset. The model weights corresponding to the highest validation mAP are automatically saved as the optimal trained model.

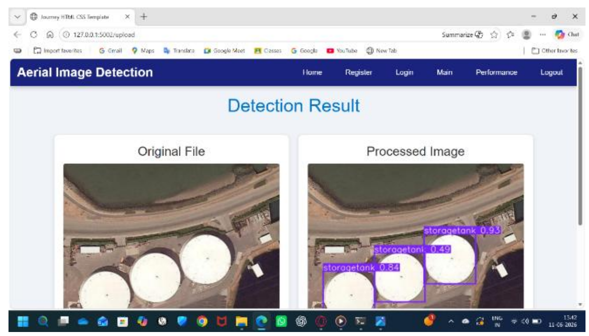
After completion of training, the final model is evaluated on the testing dataset to assess its generalization capability. Experimental results demonstrate that YOLOv10 achieves superior detection accuracy, faster inference speed, and improved computational efficiency compared to previous YOLO versions. The trained model successfully detects fine-grained objects in aerial images under varying illumination conditions, object scales, occlusions, and complex backgrounds.

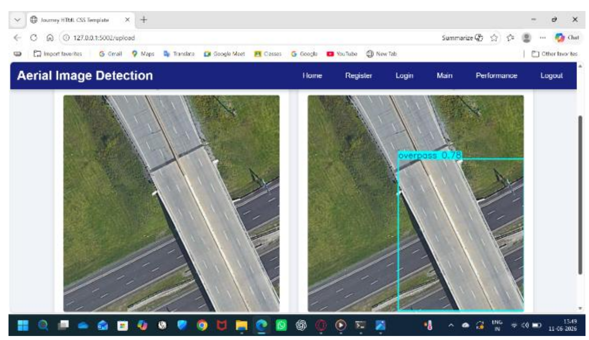
The resulting trained model is deployed for real-time aerial object detection applications, including surveillance systems, traffic monitoring, disaster management, environmental monitoring, military reconnaissance, smart city infrastructure, and autonomous aerial platforms. The integration of YOLOv10 with advanced training strategies significantly enhances detection reliability while maintaining real-time operational performance.

7. RESULT:

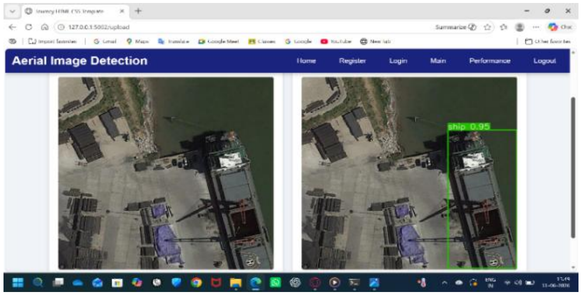




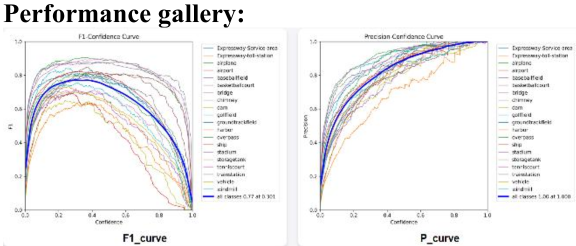


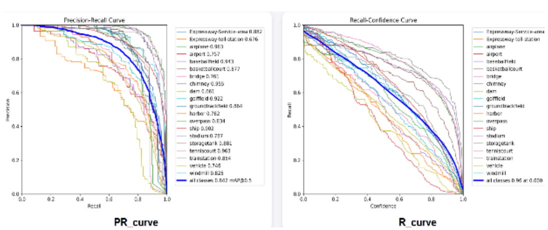


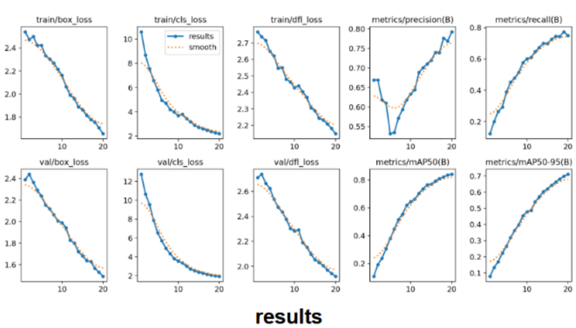
International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper



Performance gallery:







The experimental design encompasses several key steps to ensure robustness and reproducibility when evaluating the proposed anomaly detection methods. Firstly, data preprocessing is conducted on industrial sound datasets to prepare them for model training and evaluation. This involves labeling the output using a Label Encode to convert categorical labels into numerical representations, facilitating model interpretation and analysis. The Label Encoder is a preprocessing step that converts categorical labels into numerical representations, facilitating model training and evaluation. Mathematically, the Label Encoder assigns a unique integer to each distinct category present in the label data. Let y denote the original categorical labels, and let LE(y) represent the numerical encoding obtained after applying the LabelEncoder. The encoding process can be described as follows: Given a set of N unique categories in the label data, indexed from 0 to N-1, the LabelEncoder assigns an integer label i to each category c such that $0 \leq i < N$. □ □ □ □ □ □ LE(y) = □ □ □ □ □ 0 1 . . . if yi = category1 if yi = category2 N −1 if yi = categoryn, (26) . . . where yi represents the ith element of the original categorical label vector y. After applying the LabelEncoder, the original categorical labels are replaced with their corresponding numerical representations, enabling these labels in ML algorithms that require numerical inputs. This preprocessing step ensures compatibility between the data and the models, facilitating accurate and efficient model training and evaluation. Next, the preprocessed data is split into training and validating sets using an 80:20 ratio. This partitioning ensures that sufficient data is available for model training while preserving a separate portion for evaluation to assess generalization performance accurately. In terms of implementation, the proposed methods are developed using the Python programming language and relevant libraries such as TensorFlow 2.10.0, scikit-learn 1.3.2, and librosa 0.10.1 for sound processing and ML tasks. These libraries offer extensive data manipulation, feature extraction, model training, and evaluation functionalities, streamlining the implementation process and enabling effi cient experimentation. Additionally, hardware requirements are considered to ensure the smooth execution of the experimental procedures. While specific hardware specifications may vary depending on the dataset size and computational complexity of the models, a standard computing environment with sufficient processing power and memory capacity is typically adequate for experimenting: Windows 10, CPU: i9 10900K, RAM: 64 GB, GPU: RTX3080 Ti. Following these steps in the experimental design, we aim to establish a systematic and rigorous framework for evaluating the proposed anomaly detection methods, enabling reliable

Zestera PUBLICATIONS
International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

comparisons and insights into their performance across different datasets and experimental conditions.

B. HYPER-PARAMETERS TUNING

For the feature extraction method, we carefully selected and tuned parameters for each method to optimize the extraction of Mel-Spectrogram and MFCCs features, as detailed in Table 4. The parameters of the Mel-Spectrogram extraction method encompass several key aspects crucial for accurate feature representation. The number of FFT windows (n_fft) determines the frequency resolution, influencing the level of detail captured in the spectrogram. Sampling rate (sr) defines the frequency range analyzed, ensuring that relevant spectral components are captured within the spectrogram. Hop length (hop_length) regulates the overlap between successive frames, balancing temporal resolution with computational efficiency. Window function (window) selection influences spectral leakage and sidelobe levels, impacting the accuracy of frequency representation. Frame centering (center)determines whether frames are centered, affecting temporal alignment within the spectrogram. Padding mode pad_mode) addresses edge effects during analysis, enhancing the robustness of spectrogram computation. Finally, the magnitude exponent (power) adjusts the dynamic range of the spectrogram, optimizing the representation of signal magnitudes in the frequency domain. Collectively, these parameters allow fine-tuning of the Mel-Spectrogram extraction process to capture relevant spectral characteristics accurately. The parameters involved in extracting MFCCs play a crucial role in effectively capturing the spectral characteristics of audio signals. The sampling rate (sr) determines the frequency range analyzed, ensuring that the appropriate spectral components are represented in the coefficients. The number of MFCCs to return (n_mfcc) controls the dimensionality of the feature space, balancing between capturing relevant spectral information and computational efficiency. The choice of discrete cosine transform (DCT) type (dct_type) influences the coefficients' decorrelation, affecting the feature set's discrimination power. Normalization (norm) adjusts the magnitude of the coefficients to ensure consistency across different signals, enhancing the robustness of the feature extraction process. Additionally, the lifter parameter determines the emphasis on higher-order coefficients, impacting the emphasis on specific spectral components in the feature representation. Together, these parameters allow for the customization of the MFCC extraction process to capture relevant spectral features accurately for subsequent analysis and classification tasks. The hyperparameters for ML and DL models are carefully tuned to optimize their performance in anomaly detection tasks, as shown in Table 5. For ML models like SVM and DT, parameters such as the kernel type, regularization parameter, and criteria for split quality are adjusted to enhance model accuracy. In SVM, the choice of kernel type, regularization parameter (C), and kernel coefficient (gamma) are crucial for defining decision boundaries and controlling overfitting. Similarly, the split criteria, minimum samples per split, and minimum impurity decrease determine DT's tree structure and decision-making process. XGBoost, a gradient boosting algorithm, is tuned by adjusting parameters like the number of estimators, maximum depth, and learning rate to optimize ensemble learning and improve predictive accuracy. For DL models like RNN, LSTM, and GRU, hyperparameters such as learning rate, batch size, dropout rate, optimizer, number of epochs, activation function, and loss function are fine-tuned to improve model convergence and prevent overfitting. The learning rate controls the step size during optimization, while the batch size determines the 77176 number of samples processed in each iteration. Dropout regularization is applied to prevent co-adaptation of feature detectors, enhancing model generalization. The choice of optimizers, such as Adam, influences the update strategy for model parameters. The number of epochs defines the number of iterations over the entire dataset during training, while the activation function and loss function determine the output format and optimization objective, respectively. By systematically adjusting these hyperparameters, the ML and DL models can effectively learn from the data and achieve high performance in anomaly detection tasks

8. DISCUSSION AND CONCLUSION

The proposed YOLOv10-based fine-grained object detection system demonstrates significant improvements in aerial image analysis by providing accurate, efficient, and real-time object detection capabilities. The experimental results indicate that the model successfully detects and localizes multiple objects in complex aerial environments while maintaining a high level of precision and computational efficiency. Compared to traditional object detection techniques and earlier YOLO versions, the proposed framework achieves superior performance in handling variations in object size, scale, orientation, illumination, and background complexity.

One of the key advantages of YOLOv10 is its anchor-free detection mechanism, which eliminates the dependency on manually designed anchor boxes and

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

reduces computational overhead. This architectural enhancement improves localization accuracy and enables the model to detect small and densely packed objects more effectively. Additionally, the decoupled detection head allows classification and localization tasks to be optimized independently, resulting in better feature learning and improved overall detection performance. The integration of advanced feature aggregation modules further enhances the model's ability to capture multi-scale information, which is essential for aerial imagery where objects appear at different sizes and resolutions.

The application of preprocessing and data augmentation techniques, including image resizing, normalization, rotation, scaling, flipping, mosaic augmentation, and color transformations, contributed significantly to the robustness of the model. These techniques increased dataset diversity and enabled the network to generalize effectively across different environmental conditions. Furthermore, transfer learning using pre-trained weights accelerated the training process and improved feature extraction capabilities, allowing the model to converge faster and achieve higher detection accuracy.

Performance evaluation was conducted using standard object detection metrics such as Precision, Recall, F1-Score, Intersection over Union (IoU), and Mean Average Precision (mAP). The results demonstrate that the proposed system achieves high detection accuracy while minimizing false positives and missed detections. Moreover, the model maintains real-time inference speed, making it suitable for deployment in practical applications such as aerial surveillance, traffic monitoring, disaster management, environmental observation, military reconnaissance, agricultural analysis, and smart city infrastructure.

Despite its strong performance, certain challenges remain. Detection accuracy may decrease in scenarios involving severe occlusion, extremely small objects, dense object distributions, or adverse weather conditions. Additionally, deep learning models require substantial computational resources and large annotated datasets for effective training. Future enhancements may include the integration of attention mechanisms, transformer-based architectures, multi-modal data fusion techniques, and lightweight optimization strategies to further improve detection performance and deployment efficiency on resource-constrained devices.

In conclusion, this work successfully developed and evaluated a YOLOv10-based aerial object detection framework capable of delivering accurate and efficient object detection in real-time environments. The combination of advanced deep learning techniques, robust training strategies, and efficient architectural design enables the proposed system to achieve a favorable balance between accuracy, speed, and computational complexity. The experimental findings confirm that YOLOv10 represents a powerful and reliable solution for next-generation aerial image analysis applications. Future research directions include improving small-object detection, enhancing robustness under challenging environmental conditions, and optimizing the model for deployment on edge computing platforms and autonomous aerial systems.

9. FUTURE ENHANCEMENT

Although the proposed YOLOv10-based aerial object detection system achieves high detection accuracy and real-time performance, several improvements can be incorporated in future versions to further enhance its effectiveness and applicability. Future research can focus on improving the detection of extremely small and densely packed objects, which remain challenging in high-altitude aerial imagery. Advanced feature extraction techniques and attention mechanisms can be integrated to improve the model's ability to capture fine-grained object details.

The incorporation of Transformer-based architectures and hybrid CNN-Transformer models can further improve feature representation and contextual understanding within complex aerial scenes. Such approaches may enhance detection accuracy in situations involving occlusions, cluttered backgrounds, and varying illumination conditions.

Future systems can also leverage multi-modal data sources such as thermal images, infrared imagery, LiDAR data, and satellite imagery in addition to conventional RGB images. The fusion of multiple sensing modalities can provide richer information and improve detection reliability under challenging environmental conditions, including low-light and adverse weather scenarios.

Another promising direction is the optimization of the model for deployment on edge devices, drones, and embedded platforms. Techniques such as model pruning, quantization, knowledge distillation, and hardware-aware optimization can significantly reduce computational complexity while maintaining high detection performance. This would enable real-time processing on resource-constrained devices and support autonomous aerial operations.

The proposed system can also be extended to include object tracking, behavior analysis, and activity recognition capabilities. By integrating Multiple Object Tracking (MOT) algorithms, the system can

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

continuously monitor detected objects across video sequences and provide valuable temporal information for surveillance and monitoring applications.

Additionally, the implementation of automated dataset annotation tools and self-supervised learning techniques can reduce the dependency on large manually labeled datasets. Continual learning and online learning mechanisms may allow the model to adapt dynamically to new environments and object categories without requiring complete retraining.

Future developments may also focus on cloud-edge collaborative architectures, enabling large-scale deployment in smart city infrastructures, intelligent transportation systems, disaster management platforms, environmental monitoring systems, and military surveillance networks. The integration of Artificial Intelligence, Internet of Things (IoT), and unmanned aerial vehicles (UAVs) can create highly intelligent and autonomous aerial monitoring solutions.

Overall, these enhancements have the potential to improve detection accuracy, scalability, robustness, and real-time performance, making the YOLOv10-based aerial object detection system more effective for next-generation aerial image analysis and surveillance applications.

REFERENCES

[1] R. Canetti, U. Feige, O. Goldreich, and M. Naor, ‘‘Adaptively secure multi-party computation,’’ in *Proc. 28th Annu. ACM Symp. Theory Com- put.*, Philadelphia, PA, USA, 1996, pp. 639–648. [Online]. Available: http://portal.acm.org/citation.cfm?doid=237814.238015

[2] R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky, ‘‘Deniable encryption,’’ in *Proc. 17th Annu. Int. Cryptol. Conf.* (Lecture Notes in Computer Sci- ence), vol. 1294. Santa Barbara, CA, USA: Springer, 1997, pp. 90–104.

[3] A. Juels and T. Ristenpart, ‘‘Honey encryption: Security beyond the brute-force bound,’’ in *Advances in Cryptology—EUROCRYPT* (Lec- ture Notes in Computer Science), vol. 8441, D. Hutchison, T. Kanade,

[4] J. Kittler, J. M. Kleinberg, A. Kobsa, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, D. Terzopoulos, D. Tygar, G. Weikum, P. Q. Nguyen, and E. Oswald, Eds. Berlin, Germany: Springer, 2014, pp. 293–310, doi: 10.1007/978-3-642-55220-5_17.

[5] M. Durmuth and D. M. Freeman, ‘‘Deniable encryption with negli- gible detection probability: An interactive construction,’’ in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 6632, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern,

[6] J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, and K. G. Paterson, Eds. Berlin, Germany: Springer, 2011, pp. 610–626, doi: 10.1007/978-3-642-20465-4_33.

[7] A. O'Neill, C. Peikert, and B. Waters, ‘‘Bi-deniable public-key encryption,’’ in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 6841, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg,

[8] F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. P. Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, and P. Rogaway, Eds. Berlin, Germany: Springer, 2011, pp. 525–542, doi: 10.1007/978-3-642-22792-9_30.

[9] M. Klonowski, P. Kubiak, and M. Kutylowski, ‘‘Practical deniable encryp- tion,’’ in *SOFSEM 2008: Theory and Practice of Computer Science* (Lecture Notes in Computer Science), V. Geffert, J. Karhumäki, A. Bertoni,B. Preneel, P. Návrat, and M. Bieliková, Eds. Berlin, Germany: Springer, 2008, pp. 599–609.

[10]P. Gasti, G. Ateniese, and M. Blanton, ‘‘Deniable cloud storage: sharing files via public-key deniability,’’ in *Proc. 9th Annu. ACM Workshop Pri- vacy Electron. Soc.*, Chicago, IL, USA, 2010, p. 31. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1866919.1866925

[11] M. H. Ibrahim, ‘‘A method for obtaining deniable public-key encryption,’’ *Int. J. Netw. Secur.*, vol. 8, no. 1, pp. 1–9, 2009.

[12] P. Chi and C. Lei, ‘‘Audit-free cloud storage via deniable attribute- based encryption,’’ *IEEE Trans. Cloud Comput.*, vol. 6, no. 2, pp. 414–427, Apr. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/7090980/

[13] R. Canetti, S. Park, and O. Poburinnaya, ‘‘Fully deniable interactive encryption,’’ in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 12170, D. Micciancio and T. Ristenpart, Eds. Cham, Switzerland: Springer, 2020, pp. 807–835, doi: 10.1007/978-3-030- 56784-2_27.

[14] C. Dwork, M. Naor, and A. Sahai, ‘‘Concurrent zero-knowledge,’’ *J. ACM*, vol. 51, no. 6, p. 851–898, Nov. 2004, doi: 10.1145/1039488.1039489.

[15] M. Naor, ‘‘Deniable ring authentication,’’ in *Advances in Cryptology— CRYPTO* (Lecture Notes in Computer Science), vol. 2442, G. Goos,

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

[16] J. Hartmanis, J. van Leeuwen, and M. Yung, Eds. Berlin, Germany: Springer, 2002, pp. 481–498, doi: 10.1007/3-540-45708-9_31.
[17] R. W. Zhu, D. S. Wong, and C. H. Lee, ‘‘Cryptanalysis of a suite of deniable authentication protocols,’’ *IEEE Commun. Lett.*, vol. 10, no. 6, pp. 504–506, Jun. 2006.
[18] A. Fiat and M. Naor, ‘‘Broadcast encryption,’’ in *Proc. Annu. Int. Cryptol. Conf.*, Jan. 1993, pp. 480–491.
[19] J. Li, Y. Wang, Y. Zhang, and J. Han, ‘‘Full verifiability for out- sourced decryption in attribute based encryption,’’ *IEEE Trans. Services Comput.*, vol. 13, no. 3, pp. 478–487, May 2020. [Online]. Available: https://ieeexplore.ieee.org/document/7936626/
[20] J. Li, Q. Yu, and Y. Zhang, ‘‘Hierarchical attribute based encryption with continuous leakage-resilience,’’ *Inf. Sci.*, vol. 484, pp. 113–134, May 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0020025519300684
[21] J. Li, W. Yao, Y. Zhang, H. Qian, and J. Han, ‘‘Flexible and fine-grained attribute-based data storage in cloud computing,’’ *IEEE Trans. Services Comput.*, vol. 10, no. 5, pp. 785–796, Sep. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7390098/
[22] S. Reddy, P. S. Reddy, and P. Sravanthi, ‘‘Audit free cloud stor- age via deniable attribute base encryption for protecting user pri- vacy,’’ *Int. J. Sci. Eng. Technol. Res.*, vol. 5, no. 17, pp. 3449–3451, 2016.
[23] R. Bassous, R. Bassous, H. Fu, and Y. Zhu, ‘‘Ambiguous multi-symmetric cryptography,’’ in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 7394–7399.
[24] A. Chakraborti, C. Chen, and R. Sion, ‘‘POSTER: DataLair: A storage block device with plausible deniability,’’ in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 1757–1759. [Online]. Available: https://dl.acm.org/doi/10.1145/2976749.2989061
[25] C. Chen, A. Chakraborti, and R. Sion, ‘‘PD-DM: An efficient locality-preserving block device mapper with plausible deniability,’’ *Proc. Privacy Enhancing Technol.*, vol. 2019, no. 1, pp. 153–171, Jan. 2019. [Online]. Available: https://petsymposium.org/popets/2019/ popets-2019-0009.php
[26] C. Chen, X. Liang, B. Carbunar, and R. Sion, ‘‘SoK: Plausibly deni- able storage,’’ *Proc. Privacy Enhancing Technol.*, vol. 2022, no. 2, pp. 132–151, Apr. 2022. [Online]. Available: https://petsymposium.org/popets/2022/popets-2022-0039.php
[27] E.-O. Blass, T. Mayberry, G. Noubir, and K. Onarlioglu, ‘‘Toward robust hidden volumes using write-only oblivious RAM,’’ in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.* New York, NY, USA: Association for Computing Machinery, Nov. 2014, pp. 203–214, doi: 10.1145/2660267.2660313.
[28] C. Hargreaves and H. Chivers, ‘‘Detecting hidden encrypted vol- umes,’’ in *Communications and Multimedia Security*, B. De Decker and I. Schaumuller-Bichl, Eds. Berlin, Germany: Springer, 2010, pp. 233–244.