

Enhancing Transportation Security: Automated Prohibited Object Detection for Baggage Inspection Leveraging YOLO-NAS

Rohan Reddy B

Department of Computer Science and Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham Chennai-601103, India
rohanbadugula@gmail.com

SP. Chokkalingam

Department of Computer Science and Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham Chennai-601103, India
chomas75@gmail.com

Abstract—The substantial increase in demand for travel has resulted in significant congestion and prolonged wait times across various transportation sectors, prominently in the aviation industry. The primary cause of these delays can be attributed to the time-consuming manual inspection of baggage to detect prohibited objects. This predicament underscores the imperative need for the deployment of a sophisticated and efficient automated system to comprehensively identify prohibited objects in real-time. In response to this, we have developed a one-stage detection system using You Only Look Once – Neural Architecture Search (YOLO-NAS), a cutting-edge iteration within the YOLO family. An extensive experimentation was conducted on the SIXray dataset, which yielded remarkable efficacy, boasting a mean Average Precision (mAP) score of 0.924. Our research findings reveal the potential of YOLO-NAS to significantly improve the accuracy and speed of prohibited object detection in baggage, thereby mitigating delays and congestion within transportation hubs. Furthermore, our research extends to comparative analysis, pitting against its predecessor, YOLOv8, where YOLO-NAS demonstrated its superiority by surpassing YOLOv8's performance by 12%. This utilization of YOLO-NAS underscores the potential for transformative improvements in baggage inspection and serves as a testament to the continued advancement of automated detection systems within the domain of transportation security.

Index Terms—Prohibited object detection, airport security inspection, YOLO-NAS, YOLOv8, Threat objects

I. INTRODUCTION

The significance of detecting threats and contraband in public places has escalated in modern society due to the surge in human traffic and the utilization of diverse transportation methods. This has made personal safety and property security paramount, especially within transportation industry, including railways and airplanes [1, 2]. The conventional approach to X-ray baggage security screening relied heavily on manual inspections conducted by security personnel. Nevertheless, this hands-on method is vulnerable to human errors, which could be compounded by factors like fatigue and emotional stress particularly in bustling stations with large passenger flow, moreover this process would consume a significant amount of time. [3] To address these limitations, an efficient automated system for detecting prohibited items is an optimal choice for

accelerating the screening process and enhancing the accuracy. However, the development of X-ray detection models encounters substantial challenges, due to issues like the limited ability to recognize objects in X-ray images, vulnerability to variations in imaging conditions, and the presence of noise.

In recent years, the adoption of deep learning-based object detection algorithms has become prevalent across various industries. Among these, the YOLO family of single-stage algorithms has emerged as a pivotal player in X-ray object detection. YOLO's approach streamlines the process by eliminating the necessity for intricate frameworks, which makes the detection process faster. This is a notable advancement compared to previous methods utilizing some traditional machine learning models like Support Vector Machines (SVM), Bag of Words (BoW) models [4], and Random Forest, which often struggled to consistently achieve high accuracy and suffered from slower imaging speeds that fell short of practical requirements [5]. Given this backdrop, our research employs the latest iteration of YOLO, known as You Only Look Once – Neural Architecture Search (YOLO-NAS) [6], for the enhanced detection and identification of prohibited objects within X-ray images of baggage. Additionally, we conduct a comparative analysis of YOLO-NAS with YOLOv8 (version 8) [7], which is the preceding version in the YOLO family.

II. RELATED WORKS

In the contemporary world, emphasizing the significance of detecting prohibited items is essential because it acts as a cornerstone in enhancing security and safety measures. Traditionally, a variety of Machine learning methods were used for the identification of prohibited items. Turcsany et al.(2013) [4] introduced an effective BoW model with SVM and SURF features for enhanced firearms detection in X-ray baggage security images, showcasing its efficacy with extensive and diverse datasets. In another research, Franzel et al. (2012) [8] analyzed X-ray image variations and improved the dual-energy object detection approach to handle distortions and rotations, by introducing a multi-view method for enhanced detection capability. Akcay et al.(2016) [9] utilized transfer learning

with deep Convolutional Neural Networks (CNN) to classify X-ray images for security screening, with a focus on detecting handguns, while also investigating the possibility of classifying multiple objects in the same context. Xu et al.(2018) [1] used a top-down attention mechanism to enrich CNN for detecting and precisely locating prohibited items at airports. They incorporated semantic feedback, attention maps, and noise reduction techniques that yielded notable results. Li et al.(2023) [10] utilized Dual Attention Mechanism Network to enhanced the detection process by addressing challenges like small targets and overlapping objects. Velayudhan et al. (2022) [11] used a two-stage transformer model to automate baggage screening in high-risk locations, leveraging their ability to capture global features and context, helping in detection of overlapped objects. Fang et al.(2023) [12] introduced FSVM, a few-shot learning model for detecting rare contraband items from baggage scans with limited labelled data, incorporating a derivable SVM layer and a combined loss function for improved decision-making. Generative Adversarial Networks (GANs) were employed by Liu et al(2023) [13] to create synthetic X-ray inspection images, that improved the capability of object detection in security screening by providing a range of synthetic scenarios that closely resembled real-world conditions. He et al. (2023) [14] sought to enhance the detection of prohibited items in security inspections by introducing a Data Augmentation Ensemble Module (DAEM) that integrates both natural and X-ray images for training and employs data augmentation techniques through ensemble learning. Object detection models such as YOLO and De-

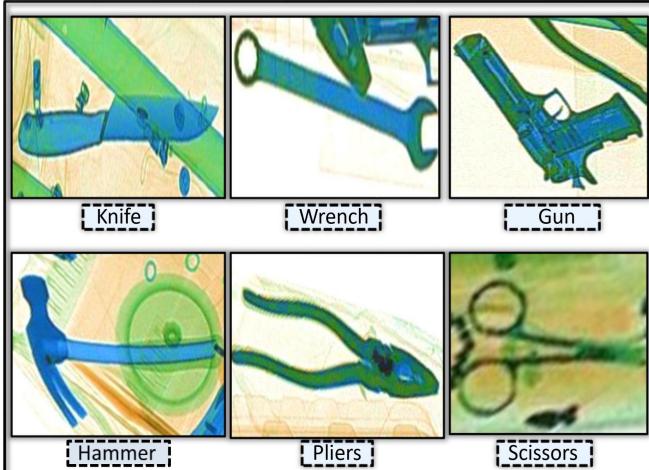


Fig. 1: Visual Depiction of Threat Items in the Sixray Dataset

tection Transformers have brought about a revolution in the field of object localization. Wei et al.(2021) [15] utilized a modified version of YOLOv3 algorithm, they incorporated Distance Intersection over Union while training and added Spatial Pyramid Pooling, which produced promising detections. Cheng et al.(2021) [3] employed the YOLOv4 algorithm to identify prohibited items in security check images for express delivery, addressing security issues involving limited-

sample, multiple-label identification. Wang et al.(2022) [16] employed YOLOv5-based network model that incorporated a transformer for enhanced feature extraction, a global attention mechanism to handle complex backgrounds and noise, and an adaptive spatial feature fusion algorithm to enhance accurate predictions. Ding et al.(2023) [17] improved the YOLOv4 model for effective detection in X-ray security images using atrous spatial pyramid pooling, an attention mechanism, optimized anchor boxes, and transfer learning. Li et al.(2023) [18] employed an enhanced YOLOv5 model, which integrates a deformable convolution module and a multi-scale feature enhancement module utilizing attention mechanisms.

III. DATA DESCRIPTION

The selection of appropriate dataset for the task of detecting prohibited objects significantly influences the model's accuracy. This accuracy is contingent upon a multitude of factors, with dataset diversity emerging as a paramount factor that significantly contributes to the robustness of the model. To address this, we have utilized the Security Inspection X-ray (SIXray) dataset [19], comprising a substantial repository of 10,59,231 X-ray images. This vast collection entails 6 distinct classes encompassing a total of 8,929 prohibited items along with the corresponding annotated files. The visual representation of these classes in depicted in Figure 1.

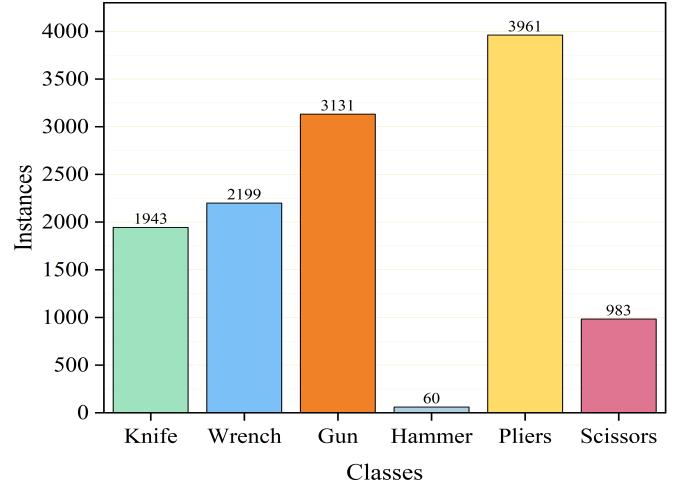


Fig. 2: Distribution of Class Instances in the Dataset

These X-ray images was derived from a multitude of subway stations, with each image being accompanied by its original metadata that indicates the presence or absence of prohibited items. The prohibited items are distributed across six common categories, namely: knives, wrenches, guns, hammers, pliers and scissors. The statistical distribution of these classes is depicted in Figure 2. Due to the limited number of available samples, the hammer class, has been excluded from our experimental analysis. Each of the images within the SIXray dataset has undergone rigorous scrutiny by security inspection machines, which have assigned distinct colors to objects composed of various materials. Furthermore, all the

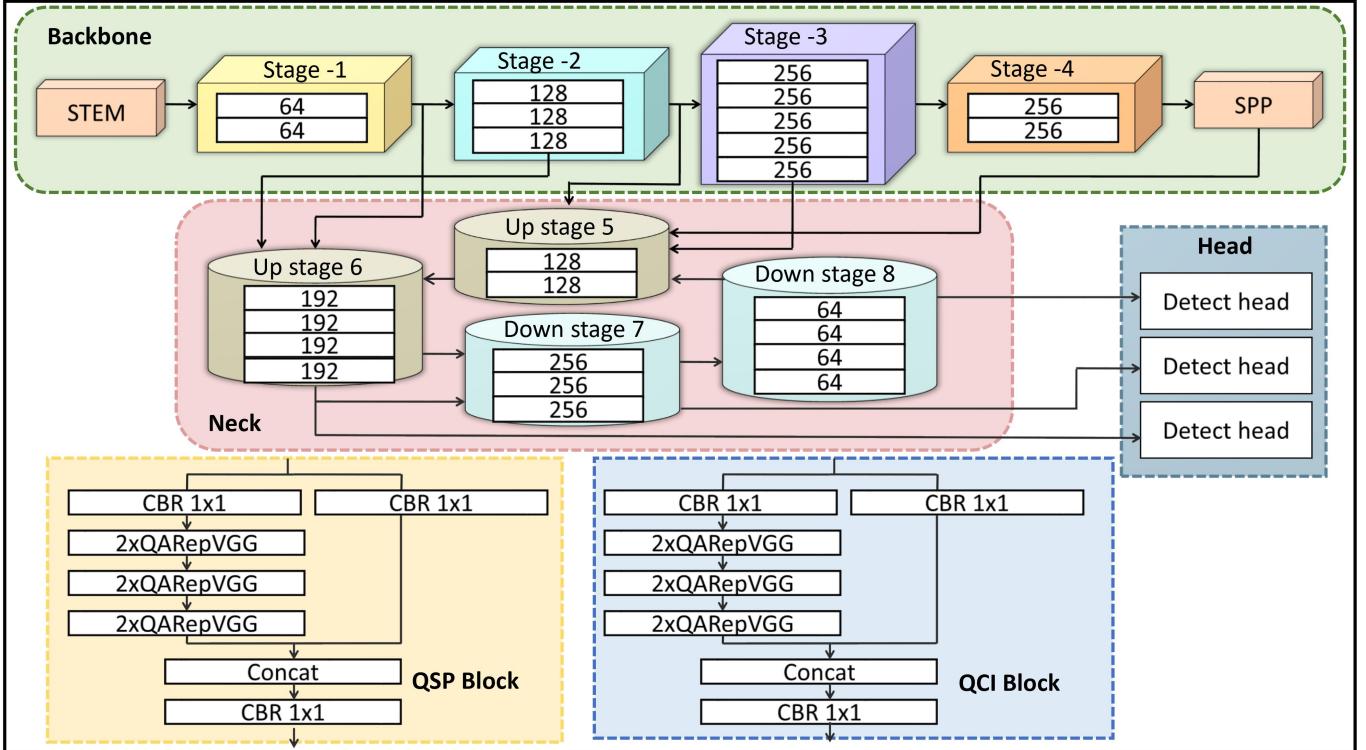


Fig. 3: Low-level schematic representation of YOLO-NAS

images have been uniformly stored in the JPEG format, with an mean pixel size of approximately 100K. In accordance with the standard practice of dataset partitioning, we have divided the original dataset, comprising 8,929 images, into a training set and validation set and test set, using a conventional 70-10-20 split ratio.

IV. METHODOLOGY

In this study, we utilized the YOLO-NAS algorithm for the purpose of detecting prohibited objects. YOLO-NAS is a groundbreaking real-time object detection model introduced by Deci.ai, excels in precision while maintaining rapid detection capabilities, effectively addressing the limitations of previous models in the series. This innovation leverages Neural Architecture Search (NAS) [20], employing optimization algorithms to autonomously design neural network architectures. This automated approach attains an optimal equilibrium between model size, computational complexity, and model accuracy, distinguishing it from its predecessors that relied extensively on manual design.

The architectural design of YOLO-NAS, relies on automated Neural Architecture Construction (AutoNAC). The exploration of the search space, leading to the revelation of YOLO-NAS's design, encompasses principles, design aspects, and computational factors. AutoNAC technology navigates through the extensive architectural search space, seeking out designs that has an optimal balance between latency and throughput. The architecture also incorporates a hybrid quantization approach, featuring the integration of Quantization-Aware RepVGG blocks [21]. This integration is aimed at ensuring the preservation of information while crafting a streamlined design suitable for edge computing applications. This hybrid quantization method introduces non-uniform quantization across the network architecture. Key facets of the YOLO-NAS structure are Post-Training Quantization (PTQ), Quantization-aware training (QAT), and Quantization Aware Blocks (QAB), namely QSP and QCI as depicted in Figure 3. Following the initial training of the neural network architecture, the model undergoes PTQ. During this phase, model size reduction is achieved by transitioning the precision of model weights from high floating-point to lower integer representations. This step aims to decrease the computational resources needed for YOLO-NAS inference while maintaining minimal accuracy loss due to the precision reduction. The resultant model proceeds to a fine-tuning process employing QAT techniques. QAT simulates the effects the impact of accuracy degradation resulting from quantization. It enables the model to acquire the ability to maintain the initial level of accuracy and to offset any loss caused to accuracy. This quantization-aware training is complemented by the architectural integration of QAB into the structure of network. These specialized blocks are designed to accommodate the consequences of reducing the precision of both weights and activations. Furthermore, within its architecture, YOLO-NAS integrates an attention mechanism. This mechanism empowers the system to prioritize regions within an image that contain the required object,

zation approach, featuring the integration of Quantization-Aware RepVGG blocks [21]. This integration is aimed at ensuring the preservation of information while crafting a streamlined design suitable for edge computing applications. This hybrid quantization method introduces non-uniform quantization across the network architecture. Key facets of the YOLO-NAS structure are Post-Training Quantization (PTQ), Quantization-aware training (QAT), and Quantization Aware Blocks (QAB), namely QSP and QCI as depicted in Figure 3. Following the initial training of the neural network architecture, the model undergoes PTQ. During this phase, model size reduction is achieved by transitioning the precision of model weights from high floating-point to lower integer representations. This step aims to decrease the computational resources needed for YOLO-NAS inference while maintaining minimal accuracy loss due to the precision reduction. The resultant model proceeds to a fine-tuning process employing QAT techniques. QAT simulates the effects the impact of accuracy degradation resulting from quantization. It enables the model to acquire the ability to maintain the initial level of accuracy and to offset any loss caused to accuracy. This quantization-aware training is complemented by the architectural integration of QAB into the structure of network. These specialized blocks are designed to accommodate the consequences of reducing the precision of both weights and activations. Furthermore, within its architecture, YOLO-NAS integrates an attention mechanism. This mechanism empowers the system to prioritize regions within an image that contain the required object,

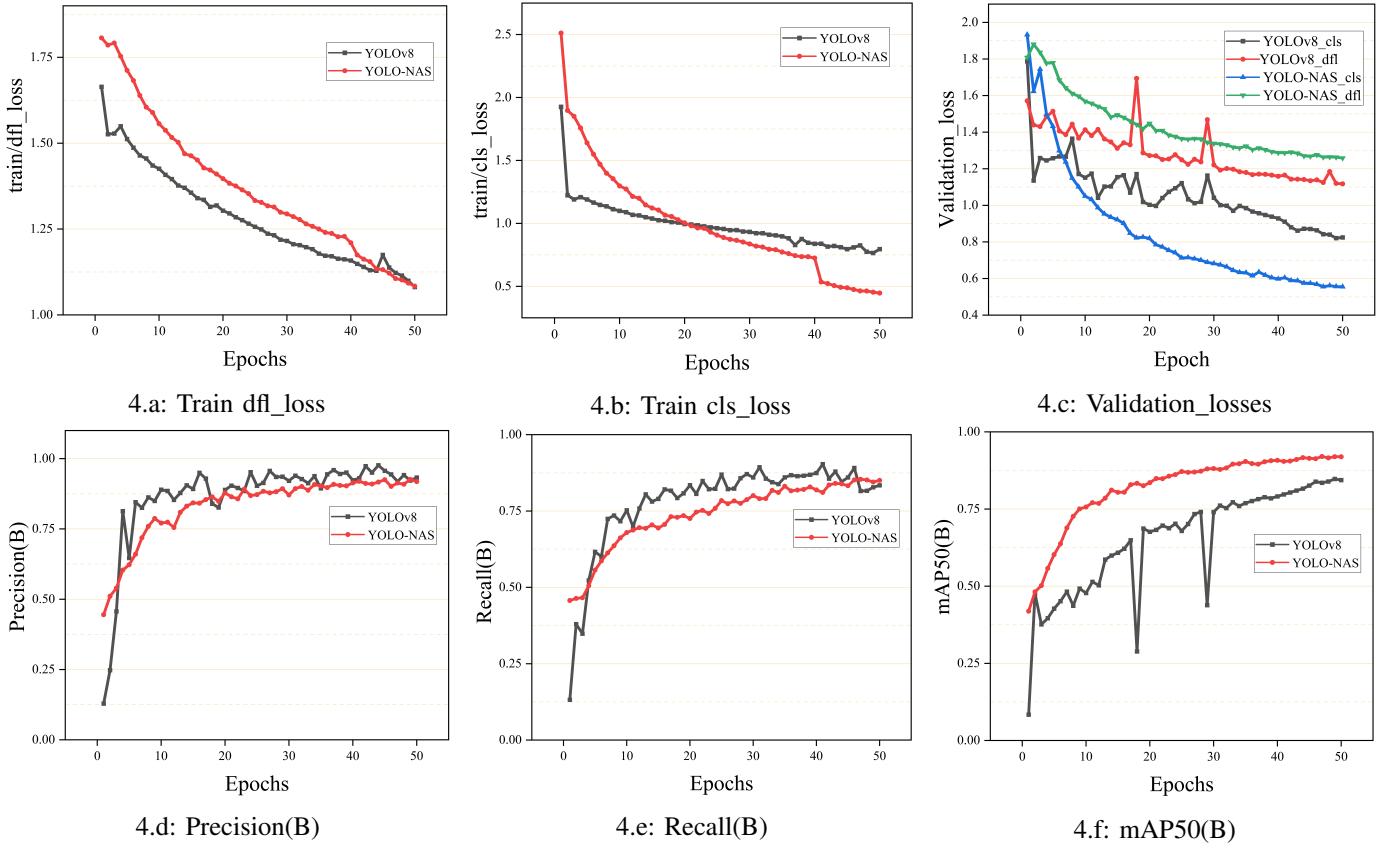


Fig. 4: Comparing Training and Validation Metrics between YOLO-NAS and YOLOv8 Models.

consequently diminishing the impact of extraneous data like background and irrelevant objects. There are three versions of YOLO-NAS, (small, medium and large).

The architecture of YOLO-NAS, is similar to its previous iterations contain three main components: the Backbone for extracting features, the Neck for merging and adjusting features, and multiple Heads for detecting objects at various scales, which synergistically construct the final framework as shown in the 5. The backbone acts as an initial feature extractor of the model, that contains STEM, four stages, starting from stage 1 to stage 4, and Spatial Pyramid Pooling (SPP) module. The stem processes the input image and the 4 stages employs QARepVGGBlock layers which gradually reduces the size of input image by performing feature extraction at different spatial resolutions. The SPP module build with Convolution layers, capture the multi-scale information. The backbone encompasses all the components to build a hierarchy of features. The Neck is responsible for enhancing feature representation and combining information from different scales. It has two sub components UpStage and DownStage modules. UpStage focuses on up- sampling features from the backbone and performing various convolutional operations, including reduction of skip connections and down-sampling. DownStage, on the other hand, is constructed to diminish the spatial dimensions of feature maps, followed by convolutional transformations. The neck contributes to feature fusion and

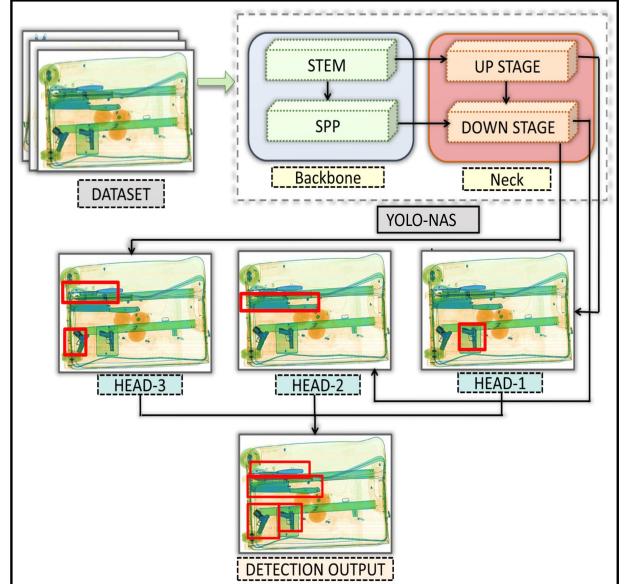


Fig. 5: Overall examination of the entire model's workflow and functioning.

scale adaptation. The head is the subsection consists three heads, where the predictions of the detection are made. Each

head is designed to predict different object scales, facilitating the detection of objects of varying sizes.

V. EXPERIMENT ANALYSIS

In our research experimentation, we aimed to evaluate and compare the performance of the YOLO-NAS model with the YOLOv8 model under identical environments, focusing on their effectiveness and robustness. Considering the substantial size of the dataset we have employed the large versions of both models, namely YOLO-NAS-1 and YOLOv8l, and conducted our experiments using a NVIDIA TESLA P100 GPU. This GPU offers superior computational capabilities, providing 1.6 times more GFLOPs and a 3-fold increase in memory bandwidth compared to the K80. We have maintained a consistent input image size of 640 x 640 pixels for both models. A comprehensive training for 50 epochs was conducted, which took approximately 13.4 hours for YOLO-NAS-1, and around 4.5 hours for YOLOv8l. Notably, our approach involved hyperparameter tuning for the YOLO-NAS-1 model, while YOLOv8l did not require explicit hyperparameter adjustments. For YOLO-NAS-1, a learning rate schedule is employed with a warm-up phase followed by a cosine annealing. The warm-up phase begins with a small learning rate of 0.000001 for 3 epochs to allow the model to stabilize initially. Subsequently, the learning rate adheres to a cosine annealing schedule, culminating with a final rate that is 10% of the initial value, which was established at 0.0005. The training process is encompassed with Adam optimizer [22] with a weight decay of 0.0001. Additionally, bias and batch normalization param-

eters have weight decay disabled during training. Exponential Moving Average (EMA) [23] is employed during training with a decay rate of 0.9. EMA helps in stabilizing the training process by smoothing the model's parameters. Mixed precision training is enabled to take hardware support for floating-point operations with reduced precision, which can speed up training without sacrificing accuracy. The PPYoloELoss loss function is utilized in conjunction with the model, which is implemented using the PaddlePaddle framework [24]. It includes class-specific regression and classification components, incorporating a dynamic anchor assignment mechanism for improved localization accuracy. The primary performance metric used to monitor the model's progress during training is mean Average Precision mAP50. This metric provides a comprehensive measure of the model's object detection capabilities. Figure 4 illustrates a set of six subplots, depicting the progression of training and validation losses and evaluation metrics including Precision, Recall and mAP50, offering a comprehensive view of the models' performance in detection as training progresses.

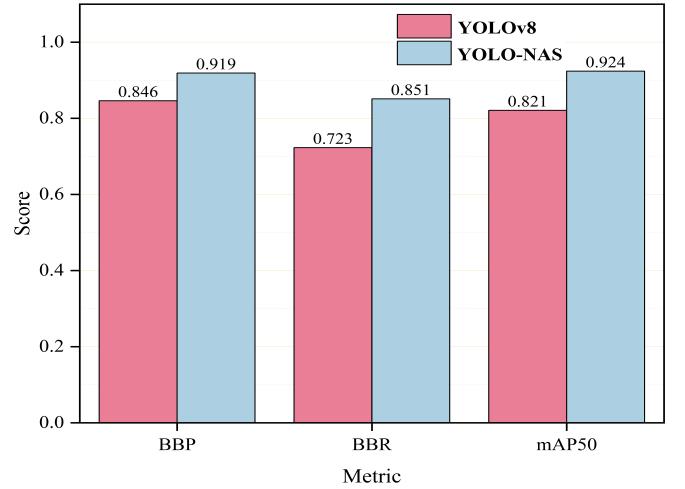


Fig. 7: Analyzing the Performance Between YOLO-NAS and YOLOv8



Fig. 6: Sample Output detection on Test Images

eters have weight decay disabled during training. Exponential Moving Average (EMA) [23] is employed during training with a decay rate of 0.9. EMA helps in stabilizing the training process by smoothing the model's parameters. Mixed precision training is enabled to take hardware support for floating-point operations with reduced precision, which can speed up training without sacrificing accuracy. The PPYoloELoss loss function is utilized in conjunction with the model, which is implemented

VI. RESULT AND DISCUSSION

In order to assess the effectiveness of our research and to make a comparative analysis of model performance, we conducted comprehensive evaluations using on the test dataset. The detected outcomes of few test images are depicted in the Figure 6. The evaluation leveraged the pertinent metrics, namely Bounding Box Precision (BBP), Bounding Box Recall (BBR), and Mean Average Precision (mAP). The BBP and BBR are computed by,

$$BBP = \frac{CPB}{CPB + IPB} \quad (1)$$

$$BBR = \frac{CPB}{CPB + MB} \quad (2)$$

Here, CPB represents the count of correctly predicted bounding boxes, IPB stands for the number of incorrectly predicted bounding boxes, and MB denotes the missed bounding boxes. The mAP serves as a comprehensive measure of prediction accuracy by computing the average precision across different object classes, providing an evaluation of a model's detection capabilities. mAP50, in particular, focuses on a crucial aspect of detection precision, accounting for predictions where the Intersection over Union (IoU) [25] between predicted and ground-truth bounding boxes exceeds a specified threshold of 0.5. The results of these metrics for the entire test dataset of

both models are presented in Figure 7, while individual class-specific metrics for YOLO-NAS model is detailed in Table I.

TABLE I: Performance Metrics of Detected Threat objects Classes

Class	BBP	BBR	mAP50
Gun	0.956	0.954	0.981
Knife	0.906	0.806	0.896
Pliers	0.937	0.856	0.931
Scissors	0.9233	0.845	0.912
Wrench	0.871	0.793	0.882

YOLO-NAS has achieved a mAP score of 92.4, surpassing YOLOv8 by a notable 12%, which recorded a mAP of 82.1. These results affirm the effectiveness of the proposed model when compared to established models. YOLO-NAS excels in object detection, offering superior accuracy while concurrently maintaining an optimal detection speed.

VII. CONCLUSION

To recapitulate, the security of baggage screening is of paramount importance across various transportation industries. In this research work we have leveraged YOLO-NAS, the latest and most effective model in the YOLO family, to detect prohibited objects in baggage during security checks. We conducted our experiments using the SIXray Dataset, known for its diversity, and achieved an impressive detection mAP of 0.924, underscoring the model's remarkable detection capabilities. Notably, the proposed model employing YOLO-NAS has surpassed the performance of all previously established methods. Its speed and accuracy hold the potential to reduce wait times at security checkpoints, benefiting numerous locations and enhancing overall security measures. Future research directions should involve working with 3D imaging to provide a clearer representation of data and exploring object segmentation techniques to further refine the detection process. Additionally, training on larger datasets with limited instances of threat objects will help our model better simulate real-world scenarios.

REFERENCES

- [1] M. Xu, H. Zhang, and J. Yang, “Prohibited item detection in airport x-ray security images via attention mechanism based cnn,” pp. 429–439, 2018.
- [2] W. Visser, A. Schwaninger, D. Hardmeier, A. Flisch, M. Costin, C. Vienne, F. Sukowski, U. Hassler, I. Dorion, A. Marciano, G. Koomen, M. Slegt, and A. C. Canonica, “Automated comparison of x-ray images for cargo scanning,” pp. 1–8, 2016.
- [3] C. Cheng, X. Wang, M. Shen, and Y. Cheng, “Image recognition of prohibited express packages based on yolov4,” pp. 1413–1418, 2021.
- [4] D. Turcsany, A. Mouton, and T. P. Breckon, “Improving feature-based object recognition for x-ray baggage security screening using primed visualwords,” pp. 1140–1145, 2013.
- [5] J. Wu, X. Xu, and J. Yang, “Object detection and x-ray security imaging: A survey,” *IEEE Access*, vol. 11, pp. 45 416–45 441, 2023.
- [6] (2023) deci. [Online]. Available: <https://deci.ai/blog/yolo-nas-object-detection-foundation-model/>
- [7] M. Hussain, “Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection,” *Machines*, vol. 11, no. 7, 2023.
- [8] T. Franzel, U. Schmidt, and S. Roth, “Object detection in multi-view x-ray images,” pp. 144–154, 2012.
- [9] S. Akçay, M. E. Kundegorski, M. Devereux, and T. P. Breckon, “Transfer learning using convolutional neural networks for object classification within x-ray baggage security imagery,” pp. 1057–1061, 2016.
- [10] Y. Li, C. Zhang, S. Sun, and G. Yang, “X-ray detection of prohibited item method based on dual attention mechanism,” *Electronics*, vol. 12, no. 18, 2023. [Online]. Available: <https://www.mdpi.com/2079-9292/12/18/3934>
- [11] D. Velayudhan, A. Hassan Ahmed, T. Hassan, M. Bennamoun, E. Damiani, and N. Werghi, “Transformers for imbalanced baggage threat recognition,” pp. 1–7, 2022.
- [12] C. Fang, J. Liu, P. Han, M. Chen, and D. Liao, “Fsvm: A few-shot threat detection method for x-ray security images,” *Sensors*, vol. 23, no. 8, 2023. [Online]. Available: <https://www.mdpi.com/1424-8220/23/8/4069>
- [13] J. Liu and T. H. Lin, “A framework for the synthesis of x-ray security inspection images based on generative adversarial networks,” *IEEE Access*, vol. 11, pp. 63 751–63 760, 2023.
- [14] J. He, Y. Zhong, B. Sun, Y. Zhang, and Y. Liu, “Data augmentation ensemble module based on natural guidance for x-ray prohibited items detection,” pp. 1–8, 2023.
- [15] Y. Wei, C. Dai, M. Chen, Z. Xu, Y. Liu, J. Fan, F. Ren, and Z. Liu, “Prohibited items detection in x-ray images in yolo network,” pp. 1–6, 2021.
- [16] Z. Wang, H. Zhang, Z. Lin, X. Tan, and B. Zhou, “Prohibited items detection in baggage security based on improved yolov5,” pp. 20–25, 2022.
- [17] B. Wang, H. Ding, and C. Chen, “Ac-yolov4: an object detection model incorporating attention mechanism and atrous convolution for contraband detection in x-ray images,” *Multimedia Tools and Applications*, pp. 1–20, 2023.
- [18] W. Li, X. Li, W. Liu, Z. Liu, J. Jia, and J. Li, “X-ray prohibited items recognition based on improved yolov5,” pp. 26–38, 2023.
- [19] C. Miao, L. Xie, F. Wan, C. Su, H. Liu, J. Jiao, and Q. Ye, “Sixray : A large-scale security inspection x-ray benchmark for prohibited item discovery in overlapping images,” 2019.
- [20] C. White, M. Safari, R. Sukthanker, B. Ru, T. Elsken, A. Zela, D. Dey, and F. Hutter, “Neural architecture search: Insights from 1000 papers,” 2023.
- [21] X. Chu, L. Li, and B. Zhang, “Make repvgg greater again: A quantization-aware approach,” 2022.
- [22] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2017.
- [23] F. Klinker, “Exponential moving average versus moving exponential average,” *Mathematische Semesterberichte*, vol. 58, pp. 97–107, 04 2011.
- [24] R. Bi, T. Xu, M. Xu, and E. Chen, “Paddlepaddle: A production-oriented deep learning platform facilitating the competency of enterprises,” pp. 92–99, 2022.
- [25] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression,” pp. 658–666, 2019.