UDC: 628.3 https://doi.org/10.15407/jai2023.02.042

IMPROVING THE ACCURACY OF LANDMINE DETECTION USING DATA AUGMENTATION: A COMPREHENSIVE STUDY

O. Kunichik[1], V. Tereshchenko[2]
[1,2]Taras Shevchenko National University of Kyiv, Ukraine
60, Volodymyrska st., Kyiv, 01033
[1]ao669517126@gmail.com
[2]V_ter@ukr.net
[1]http://orcid.org/ 0000-0002-4938-9446
[2]http://orcid.org/ 0000-0001-7639-2969

Abstract. In areas such as landmine detection, where obtaining large volumes of labeled data is challenging, data augmentation stands out as a key method. This paper investigates the role and impact of different data augmentation methods, and evaluates their effectiveness in improving the performance of deep learning models adapted to landmine detection.

Landmine detection is governed by international security requirements on the one hand, and urgent humanitarian needs on the other. This field, characterized by its urgency and the requirement for meticulous accuracy, is key against the explosive ordnance. The hidden dangers of these munitions go beyond direct physical damage, leaving their mark on the socio-economic structures of the affected regions. They hinder agricultural activities, impede the restoration of infrastructure and create obstacles to the return and resettlement of displaced populations. The mission to detect and neutralize these hidden hazards combines advanced technology with an unwavering commitment to humanitarian principles to leave future generations with a land cleared of the heavy legacy of past wars.

The effectiveness of machine learning models in detecting landmines is inextricably linked to the diversity, volume and reliability of the data they are trained on. The effort to collect a diverse and representative dataset is fraught with challenges, given limitations related to accessibility, ethical considerations and security issues. The lack of comprehensive data poses significant obstacles to the development and refinement of machine learning algorithms, potentially limiting their ability to operate effectively in diverse and unpredictable areas.

In response to these limitations, data augmentation has become an important method. It is a way to circumvent data limitations by supplementing existing datasets with synthesized variations. Augmentation strategies include spatial alignment, pixel intensity manipulation, geometric transformations, and compositing, each of which is designed to give the dataset a semblance of real-world variability.

This study explores the various applications of data augmentation in the field of landmine detection. It emphasizes the importance of augmentation as a means of overcoming data limitations.

Keywords: Landmine Detection, Data Augmentation, Machine Learning, Dataset Enhancement, Computer Vision, Deep Learning Architectures.

1.Introduction

Landmine detection plays a key role in global security and humanitarian efforts, ensuring the safety of people in war-torn areas. Detecting these often invisible threats is a process accompanied by many challenges, one of the most important of which is the lack of reliable and diverse data suitable for training pattern recognition systems. This article discusses the importance of landmine detection, the challenges associated with limited datasets, and explores an innovative solution for data augmentation to improve detection capabilities.

Landmine detection is not only a technical challenge; it has profound humanitarian implications. Undetected landmines continue to pose risks that result in casualties, hindering socio-economic development and impeding post-war recovery and the return of people to their homes. For example, the de-occupied territories of Ukraine are a continuous zone of contamination by landmines and other explosive hazards [1]. Therefore, effective landmine detection systems are becoming essential to ensure both human safety and the rapid recovery of the affected areas.

Modern landmine detection relies heavily on algorithmic approaches, such as machine learning models, which require diverse and comprehensive datasets to perform optimally [2], [3]. However, obtaining such datasets is challenging. Conflict zones, which are often prime locations for data collection, pose logistical, ethical, and geopolitical obstacles that make data collection limited and difficult. This scarcity impedes the development of robust algorithms, leading to the risk that models will not generalize and will not be effective across different territories.

To address the challenges posed by data scarcity, data augmentation emerges as a promising approach. This technique amplifies both the volume and diversity of datasets through artificial means. Employing a range of transformations, such as spatial, pixel-based, and temporal (spanning day-night shifts), data augmentation enriches the quality and scope of training data. This not only curtails the potential for model overfitting but also equips models to adapt to real-world variability, enhancing the accuracy of landmine detection.

In the realm of pattern recognition, augmentation serves as a pivotal instrument to enhance data utilized in machine learning, especially deep learning. Through diverse transformations, including rotation, scaling, cropping, flipping, and noise addition, it bolsters data diversity and quality. These modifications are vital for elevating model precision and recall rates. This manuscript offers an overview of augmentation methodologies employed within a broader project dedicated to constructing an explosive ordnance detection system [4].

2.Related Work

Different types of images and tasks require specialized augmentation methods. To this end, many studies have developed frameworks and libraries to provide a wide range of image augmentation methodologies. Paper [5] made a significant contribution to a broad overview of image data augmentation methods, assessing their impact on the main tasks of computer vision, namely semantic segmentation, image classification, and object detection.

The imgaug library [6] contains many methods, such as flipping, rotation, noise addition, contrast change, and others, which are used in the study. Also, in [7], the "Keras preprocessing layers" were introduced, a module integrated into TensorFlow that facilitates image resizing, scaling, rotation, flipping, and other augmentation processes. This paper also includes a practical guide that explains how to use these layers to process datasets and train models.

Among recent developments, the "albumentations" library [8] deserves special attention. This library offers an efficient and flexible tool for image augmentation, presenting a variety of methods optimized for various computer vision tasks. The flexibility and extensibility of "albumentations" position it as an essential asset for researchers and practitioners in this field.

3.The need to supplement the detection of landmines: Overcoming dataset limitations and issues of overfitting

In the complex field of landmine detection, collecting comprehensive datasets is a huge challenge, which emphasizes the indispensable role of data augmentation. The foundation of effective landmine detection models is a dataset that reflects the diverse typologies of landmines scattered across a range of terrains, atmospheric conditions and types of emplacements. However, the effort to assemble such a comprehensive collection faces pragmatic obstacles. The search for authentic, multifaceted images of landmines faces many logistical, ethical and security challenges. The lack of diverse images of landmines poses a huge obstacle, making it difficult to develop models that are universally adaptable.

Against this backdrop, augmentation is a reasonable solution. By skillfully applying a variety of transformations to existing images, augmentation artificially increases the diversity in a dataset. This careful process produces a dataset that, while based on a limited set of authentic samples, resonates with the unpredictability and complexity of real-world landmine encounters.

Limited datasets invariably raise the spectre of overfitting, a phenomenon where models, in their quest for accuracy, become constrained by the specifics of the training data, decreasing their effectiveness in new scenarios. The lack of real landmine imagery exacerbates this problem. Without sufficient variability, models tend to memorize the features of the dataset, which makes them poorly adapted to real-world conditions.

This is where augmentation comes in. By generating many synthetic variations based on a base set, it effectively expands the variability of the model. This augmentation reduces the risks associated with overfitting the model, contributing to models that, although based on limited real-world data, are able to recognize the diverse environmental combinations associated with landmines.

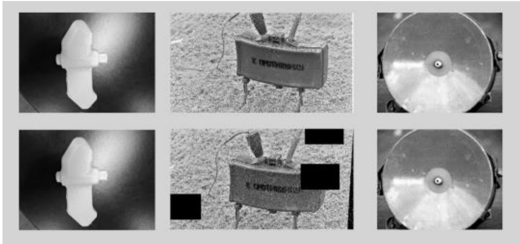
4.Common augmentation methods: Exploring the complexities of data augmentation in landmine detection

4.1. Basic augmentation techniques

Landmine detection benefits greatly from data augmentation, which uses a set of techniques to enhance and diversify the dataset. This section focuses on the main types of augmentation techniques relevant to this field: spatial transformations, pixel-level variations and geometric changes. The impact of different techniques on different objects may vary. Determining which algorithm to apply to an object is learned through experience and experimentation. For example, grayscale for some types of mines (round MON-100 and MON-200) (Fig.1.c) significantly reduces the accuracy of the models, while for others, such as PFM-1 (petal) (Fig.1.a), it increases it. This is because the former, when grayscaled, becomes simple round objects, while for the petal, which has a wide range of colors, this, on the contrary, helps to improve accuracy. For MON-50 grayscale is an option - it can be different colors (Fig.1.b).

4.1.1. Spatial transformations

Spatial transformations change the overall arrangement of an image without changing its content. The most common methods include rotation, scaling, cropping, and flipping. Rotation provides different angles of the same image (Fig. 2). Zooming allows you to get a close-up or wide view. Cropping focuses on specific parts, and flipping creates mirror images, adding variety to the dataset.



(a) (b) (c)

Fig. 1. Grayscale: PFM-1 (a), MON-50 (b), MON-100 (c)

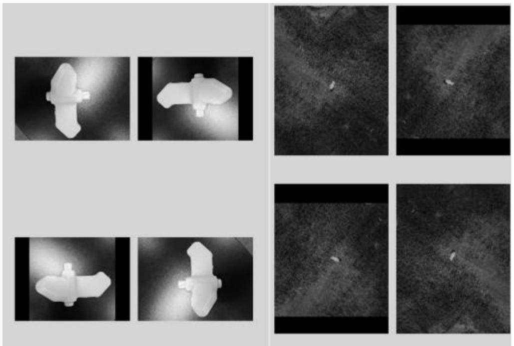


Fig. 2. Rotate PFM-1

4.1.2. Variations at the pixel level

Pixel-level adjustments adjust brightness, contrast, saturation, and even introduce noise (Fig. 3). These adjustments help models train on images that simulate different lighting conditions and minor imperfections that are common in the real world.

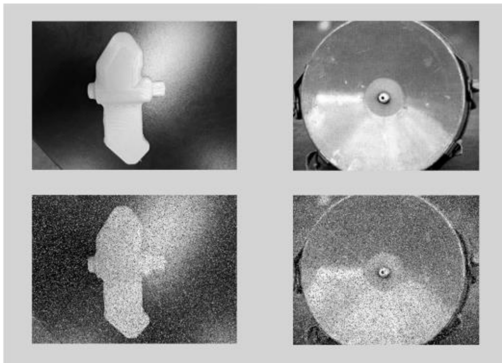


Fig. 3. Noise 25% PFM-1 (left), MON-100 (right)

4.1.3. Geometric and morphological transformations

Geometric alterations involve manipulating an image to distort its structure, such as stretching or curving it. Morphological techniques, such as dilation and erosion, shear (pic. 4), change the contours and features of an image. Both types help models to recognize landmines in different terrains and under different conditions.

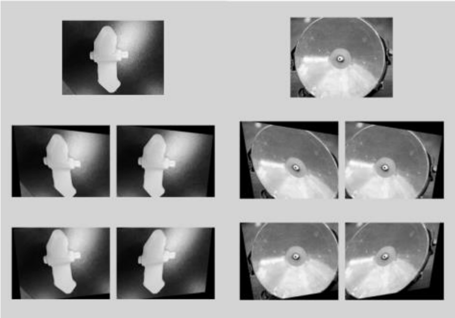


Fig. 4. Shear PFM-1 (left), MON-100 (right)

Thus, these augmentation techniques expand and diversify the training data. By simulating different conditions and scenarios, they prepare models for real-world challenges in landmine detection, increasing accuracy and reliability.

4.2. Advanced Augmentation Techniques

In the study, advanced data augmentation holds a pivotal position. These techniques are integrated into the YOLOv8 training process, enhancing the data's variety and subsequently the model's performance. Let's introduce definitions of some metrics.

In machine learning, the term "loss" refers to a measure of how well a model's predictions match the true values. There are many different loss functions, such as Mean Absolute Error (MAE), Mean Squared Errors (MSE), Sum of Squared Errors, etc. The latter is mathematically expressed by the formula:

$$L_{SSE}(y,\hat{y}) = \sum_{i=1}^{n}(y_i - \hat{y_i})^2,$$

where y is the true value and $\hat{y}$ - is the predicted value. A larger loss, or also error, indicates a larger discrepancy between the predictions and the true values.

The Box loss is the specific metric that measures how close the predicted bounding box is to the actual labels on the image in the dataset. In YOLO the Mean Square Error loss function is used to calculate the Box loss [15]:

$$L_{MSE}(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2,$$

where y is the true value and $\hat{y}$ - is the predicted value.

The Class loss is calculated on the Binary cross-entropy loss (or Log-loss) function for the confidence values of each bounding box between predicted and ground truth ones:

$$L_{BCE}(y,\hat{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i \log\hat{y_i} + (1-y_i)\log(1-\hat{y_i})),$$

where y is the true value and $\hat{y}$ - is the predicted value.

Box loss is usually understood as the difference between the predicted coordinates of the object's bounding box and the actual coordinates of the bounding box. In contrast, cls_loss quantifies the difference between the predicted class labels and the true class labels.

With mosaic and mix-up augmentations activated during YOLOv8 training, we have noted elevated values for `box_loss` and `cls_loss`. This is due to the nature of the Mosaic method – it combines 16 images from a dataset, and Mix-up makes these pictures merged from several files [9]. That is why the box loss and the class loss in these augmentation methods becomes higher with increasing precision and, particularly, recall (Fig 5-6). However, when these parameters are turned off, their values are significantly reduced to less than 0.01. It should be noted that even with these loss values, the precision and recall remain very high – both exceed 90%.

Maintaining high precision (1) and recall (2) remains crucial so it is acceptable not to pay attention to high `box_loss` and `cls_loss` metrics.

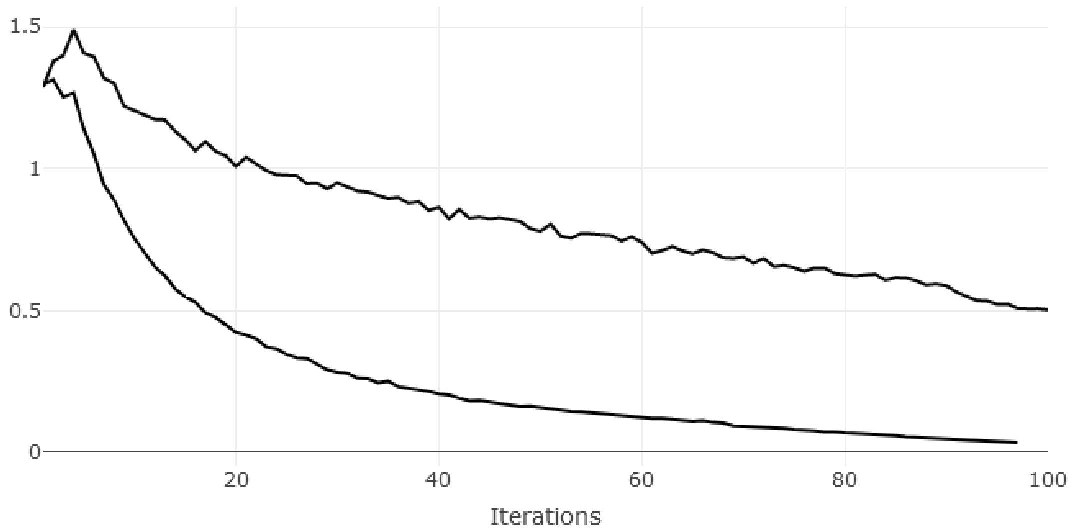


Fig. 5. Box loss for training with Mosaic and Mix-up (top line) and without them (bottom line)

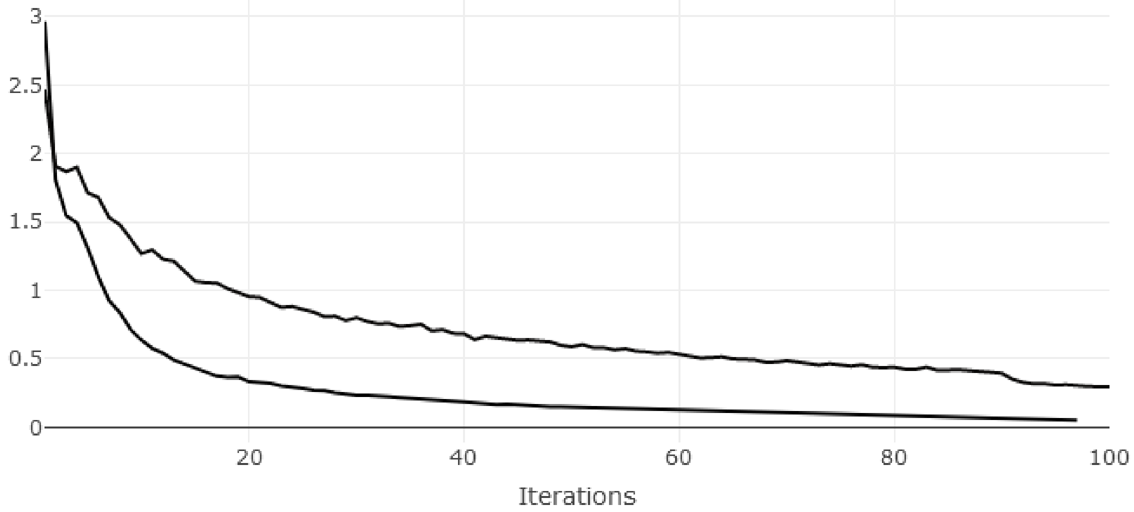


Fig. 6. Class loss for training with Mosaic and Mix-up (top line) and without them (bottom line)

Figures 5-6 show the last 10 epochs of the learning processes when Mosaic was disabled (the default YOLOv8 setting), the upper lines on Fig. 5-6 go down because there are no combined images in the training process (Fig. 8).

Precision is an indicator of how often the model's predictions are correct, and recall indicates how many true alarms were identified by the model (Fig. 7 and formulas (1), (2)).

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives+False Positives}} \quad (1)$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives+False Negatives}} \quad (2)$$

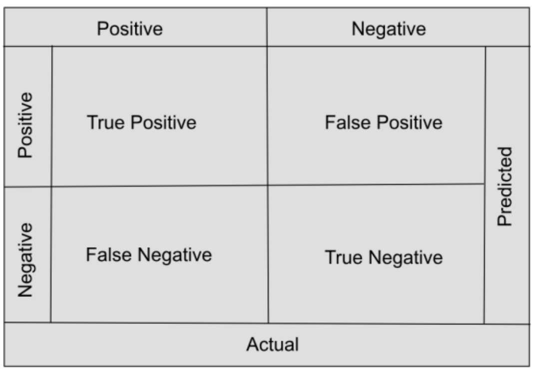


Fig. 7. The Confusion Matrix

Balancing these metrics, as well as managing box_loss and cls_loss, is vital to achieving optimal performance, especially in tasks such as object detection.

Here, some advanced strategies that add depth and adaptability to the data are explored. The current study uses algorithms from the YOLO family. So, one of the methods used by default is a Mosaic - a set of several images grouped into a single image.

4.2.1. MixUp and CutMix

MixUp and CutMix [9, 10] are the techniques that go beyond simple image modification. They combine parts of different images and their labels. This not only diversifies the labels, but also provides models with a wider selection of images to learn from. This approach helps the models understand different types of landmines and reduces the likelihood of false positives. In the study, we use Mix-Up together with Mosaic (Fig. 8).

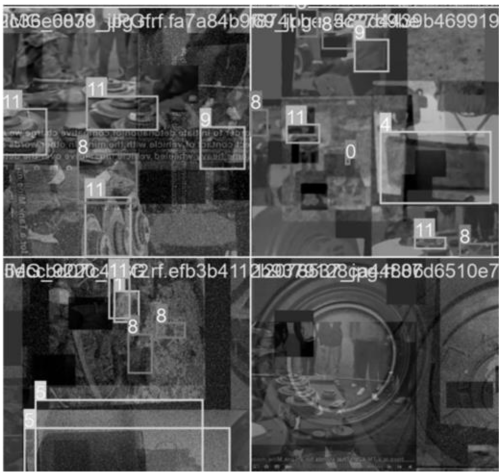


Fig. 8. The Part of the mosaic of mix-ups

4.2.2. GAN-based Augmentation

Generative adversarial networks (GANs) [11] have reshaped the perspective on data augmentation. They are adept at producing images closely resembling actual mines. A GAN is structured with two components: a generator, which crafts images, and a discriminator that evaluates their authenticity. This interplay aids models in deepening their understanding of landmine appearances. The inclusion of these synthetic images in the dataset enriched the training examples of the models. This approach is earmarked for implementation in upcoming study phases.

4.2.3. Sim2Real augmentation

Sim2Real [12] combines virtual simulations with real data. These simulations contain a diverse set of scenarios and challenges, allowing the models to learn from both simulated and real environments. The main benefit is the enhanced ability of the models to identify landmines under different conditions, surpassing the limitations of simple camera snapshots. Although we have considered this method, it has not yet been integrated into research.

In summary, by applying these advanced techniques, it is possible to manage diverse and complex data sets for the models. This enriched data bolsters the precision and adaptability of the models. Such strategies redefine the potential in landmine detection, enhancing the efficacy and safety of the solutions.

5.Experiment and results: Testing the preprocessing methods

In this section, we will discuss the different data preprocessing methods we used and how they affected the performance of the model.

While the primary focus of the study is on data augmentation, it's crucial to touch upon the initial steps of preprocessing. Although preprocessing doesn't increase the dataset size like augmentation, it remains a foundational phase in most machine learning processes. One such integral process is resizing all images to maintain consistency across the dataset. The study recognized and used numerous pre-processing tools to improve data quality. Specifically:

– Auto-Orient was used to standardize image orientation, ensuring uniformity in model input.

– Resizing all images provided a consistent dimension, ensuring dataset consistency.

– Leveraging the auto-adjust contrast ensured clearer, more discernible images, facilitating improved pattern detection by the models.

– While it was initially considered converting all images to grayscale, later it was opted to augment only 30% of the dataset in this manner, as it yielded superior outcomes.

The repercussions of these preprocessing strategies on the model's effectiveness are elaborated upon in the provided Table 1.

Table 1. Results of experiments with preprocessing

Version ID	mAP / Recall /Precision	Preprocessing	Augmentations
6	94.4/82.4/93.6	Auto-Orient, Resize: Stretch to 640x640	No
17	89.1/83.0/91.3	Auto-Orient, Resize: Stretch to 640x640, Auto-Adjust Contrast	Grayscale: Apply to 30% of images
19	91.0/84.2/93.4	Auto-Orient, Resize: Stretch to 640x640, Auto-Adjust Contrast	Grayscale: Apply to 30% of images., Cutout: 3 boxes with 21% size each
20	88.8/81.4%/91.2	Auto-Orient, Resize: Stretch to 640x640, Auto-Adjust Contrast	Grayscale: Apply to 40% of images
21	89.6/81.1/92.1	Auto-Orient, Resize: Stretch to 640x640, Auto-Adjust Contrast	Grayscale: Apply to 40% of images., Cutout: 3 boxes with 21% size each
26	88.7/84.6/89.0	Auto-Orient, Resize: Fit (white edges) in 640x640	Grayscale: Apply to 30% of images. Cutout: 3 boxes with 21% size each
27	89.5/82.1/91.7	Auto-Orient, Resize: Fit (black edges) in 640x640	Grayscale: Apply to 30% of images. Cutout: 3 boxes with 21% size each
29	91.3/85.9/90.3	Auto-Orient, Resize: Fit within 640x640	Grayscale: Apply to 30% of images. Cutout: 3 boxes with 21% size each
39	91.2/85.2/91.7	Auto-Orient, Resize: Fit within 640x640, Auto-Adjust Contrast: Using Contrast Stretching	Grayscale: Apply to 30% of images. Cutout: 3 boxes with 21% size each
41	94.2/90.2/96.1	Auto-Orient, Resize: Fit within 640x640, Auto-Adjust Contrast: Using Contrast Stretching	Grayscale: Apply to 30% of images. Cutout: 3 boxes with 21% size each
42	93.7/89.4/95.2	Auto-Orient, Resize: Fit within 640x640, Auto-Adjust Contrast: Using Contrast Stretching, Flip: Horizontal, Vertical	Grayscale: Apply to 30% of images. Cutout: 3 boxes with 21% size each

6.Dataset Overview: Utilizing YOLOv5 and Roboflow [13]

We started with a diverse collection of landmine photographs. This collection of different types of landmines captured under different conditions laid the foundation for the experiments. Our initial modifications to the data were done on the Roboflow platform, where the model was also published [16]. Several augmentations were applied here, including grayscale, cutout, rotation, flip, shift, blur, and noise, adapted specifically for the YOLOv5 model. The best results were obtained with Cutout 21% and Grayscale (Table 1, Version Id 41). At this stage, we switched to the more modern YOLOv8 model and tested different augmentation techniques again. After testing different configurations, the following techniques were selected, as shown in the Table. 2.

These methods were chosen based on the qualitative performance of each method applied to the same dataset, and the metrics of all experiments are shown in Table 3 (There are all experiments listed – for stages 1 and 2).

Table 2. The best methods of augmentation
on the first stage

#	Augmentation
1	Flip: Horizontal, Vertical
2	90° Rotate: Clockwise, Counter-Clockwise, Upside Down
3	Grayscale: Apply to 30% of images
4	Noise: Up to 15% of pixels

Table 3. The metrics of methods of augmentation on the 1 and 2 stages

Augmentation Technique(s)	Precision	Recall	mAP50	mAP50-95	Fitness
No 1 stage and ALL 2nd stage(YOLO) aug	99	89.2	95.3	76.7	78.6
Grayscal, Rotate, Noise, Flip, ALL 2nd stage augment	97.4	92.6	95.7	77.3	79.2
Grayscal,Rotate, Noise, Flip. No 2nd st.	96.6	79	89.1	68.1	70.2
Grayscale No 2nd st.	96.2	68.9	84.6	66.1	68
Noise No 2nd st.	93.3	74.9	86.4	64.7	66.9
Rotate No 2nd st.	90.8	73.3	86.6	65.7	67.8
Flip No 2nd st.	90.2	76.1	86.9	68.4	70.2
Bounding Box Rotate No 2nd st.	90.1	70.5	83.6	63.3	65.4
Mosaic No 2nd st.	91.6	71.8	82	61.3	63.4
No augmentation	91.6	71.8	82	61.3	63.4
Blur No 2nd st.	87.6	75.1	83.8	61.6	63.8
Bounding Box Rotate No 2nd st.	88	75.8	85.3	59.6	62.2
Cutout No 2nd st.	86	63.2	80.2	61.5	63.3

The second stage: Switching to YOLOv8

When YOLOv8 was selected for training, the best practices from the first phase were used and additional augmentations were implemented, as shown in tab. 4. Results also could be found in the tab. 3.

Table 4. The YOLOv8 augmentation parameters

Augmentation Method	YOLO Code	Description
Mosaic	mosaic	Create a mosaic of four images
HSV Hue Shift	hsv_h	Shift hue in HSV color space
HSV Saturation Shift	hsv_s	Shift saturation in HSV color space
HSV Value Shift	hsv_v	Shift value in HSV color space
Degrees Rotation	degrees	Rotate images by specified degrees
Translate	translate	Translate images by specified values
Scale	scale	Scale images by specified factors
Shear	shear	Apply shear transformations
Flip Vertical	flipud	Flip images vertically
Flip Horizontal	fliplr	Flip images horizontally
Mixup	mixup	Apply mixup to combine images

At this point, it worth to mention the mixing and mosaic techniques. It is worth noting that mosaic gave the best results among all the methods that were tested. The Table 3 shows that Blur and Bounding Box Rotate, although they give lower precision, increase the recall, the best methods applied together (Grayscale, Rotate, Noise, Flip) give the maximum result, and when paired with the above-mentioned techniques from stage 2 (Tab. 4), the best result was achieved with an precision of 97.4 and a recall of 92.6. Although the experiment with only stage 2 augmentation is on the first place in the table, the recall is much lower, so the following methods were considered: Grayscale, Rotate, Noise, Flip, All 2-nd stage augmentations set to be the best model.

The following notations are worth noting:

– **Average Precision** (AP) is a metric that calculates the precision of an object detection algorithm for a specific class. It is calculated as the mean of the precision at various recall levels, generally visualized using a precision-recall curve. The formula is represented as the area under the precision-recall curve, typically computed as:

$$AP = \int_0^1 p(r)dr$$

where p and r are precision and recall, which are calculated using formulas (1) and (2).

– **Mean Average Precision** is calculated as follows:

$$mAP = \frac{1}{n}\sum_{k=1}^{n} AP_k$$

where AP_k is Average precision of a class k.

– **mAP50:** Mean Average Precision at 50% IoU (Intersection over Union). IoU measures the overlap between two bounding boxes. mAP50 is the mean of the average precision scores at IoU of 50%.

– **mAP50-95:** This is the mean average precision calculated at different IoU thresholds from 50% to 95%. It's a more rigorous metric than mAP50 as it averages mAP over a range of IoU values.

– **Fitness:** a value that YOLO defaults to a weighted combination of metrics: mAP@0.5 with 10% weight, and mAP@0.5:0.95 with 90%. In the Table 3 it can be observed that the Fitness metric is the highest for 2 row that we chose as the best.

Model training and results

Using the YOLOv5 model and later the YOLOv8 model for recognition, the models were trained on augmented data from both stages. The combination of different augmentations ensured that the models were exposed to a wide range of variations, which contributed to better generalization. As a result, it can be observed a big jump compared to the data without augmentation and with augmentation.

6.1. Progressing to YOLOv8

Transitioning to YOLOv8 for further training, we blended the top methods from the previous phase and introduced new augmentation processes. In this phase, a strong reliance was placed on the mixing and mosaic techniques, with the latter demonstrating the most promising results in the evaluations conducted.

For example, in Fig. 9 it can be observed that the lines are arranged in ascending order for the Precision of Cutout, Greyscale, and methods delineated in Table 2, as well as the combined methods from Tables 2 and 4 (the top line).

The better difference for recall (same line order as in Fig. 9) in Fig. 10 is the main reason why we chose the 2nd set of augmentations from Table 3. Since the recall metric given in (2) plays a crucial role for landmine detection, the precision can be lower if the recall increases significantly. Simply put, it is acceptable that not all detected landmines are landmines (lower precision), but it is very important not to have objects that are landmines but were not detected as landmines at all.

6.2. Training Process and Findings

Our training kicked off with the YOLOv5 model, moving later to YOLOv8. We harnessed data enriched with variations from both the initial and advanced phases. This diverse exposure allowed the models to experience a vast array of data changes, resulting in more adaptable models. The stark improvement was evident when comparing the graphs from Figures 9-12.

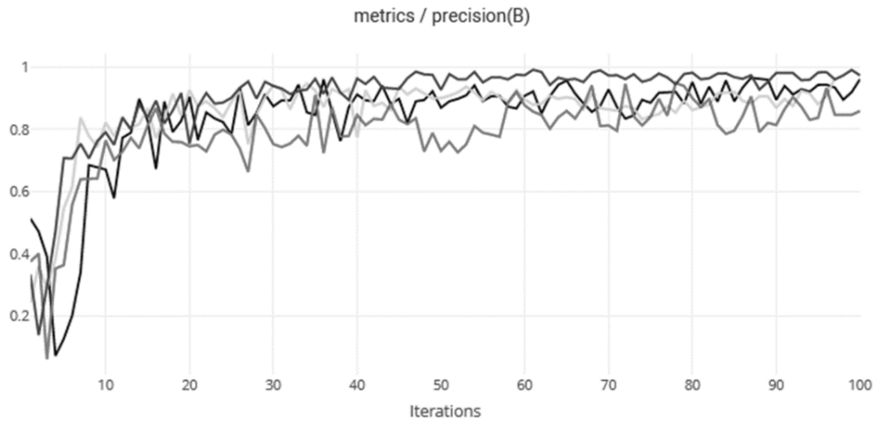


Fig. 9. Precision of Cutout, Greyscale, Methods from Table 2 and Methods from the Tables 2 and 4

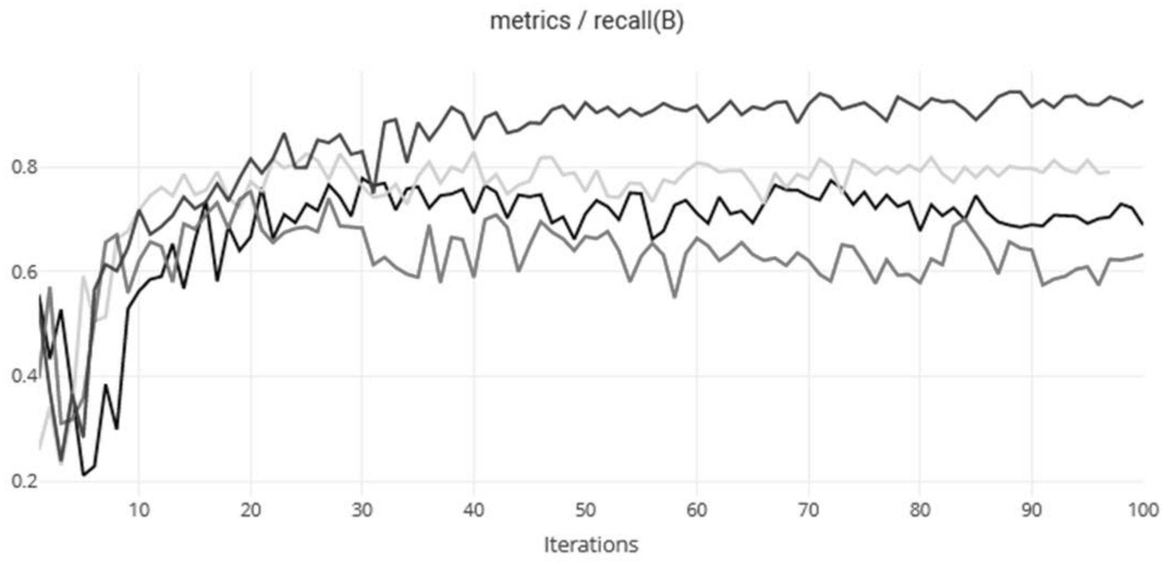


Fig. 10. Recall graph Also the same order of lines for mAP@50 and mAP@50-95 given in Fig. 11-12

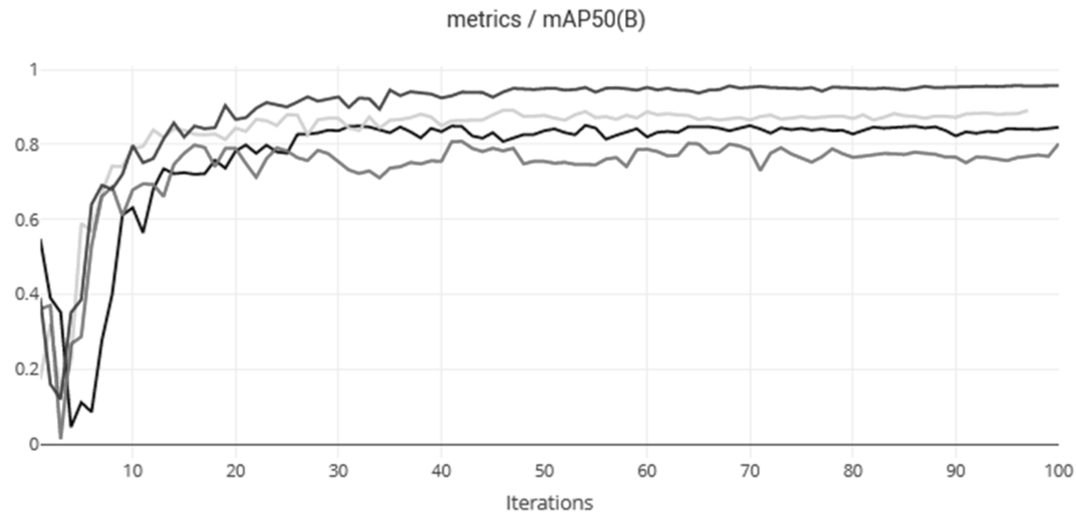


Fig. 11. mAP@50 graph

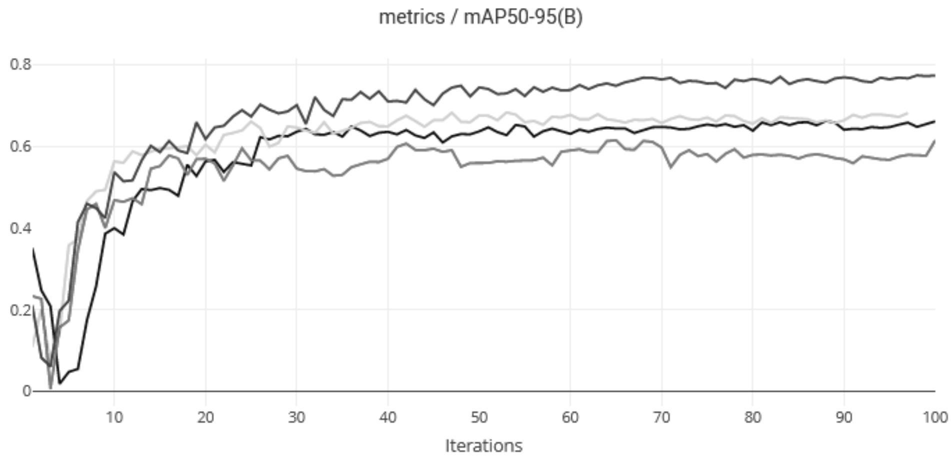


Fig. 12. mAP@50-95 graph

The experiments with different types of augmentation emphasized the importance of data diversity when training robust models. Mosaic, blending, and other methods from Phase 2 (Table 4) proved that augmentation can significantly improve important metrics such as recall. Our findings pave the way for further research into advanced augmentation techniques to improve landmine detection.

7. Challenges and insights in landmine detection through data augmentation

There are both benefits and challenges to using data augmentation for landmine detection. Appropriate application of these techniques is essential to ensure the accuracy of the model and its application in the field.

7.1. Unnatural scenarios

Data augmentation can inadvertently lead to the creation of images that do not reflect real-world mine risk scenarios. For example, converting images to grayscale may improve certain characteristics, but it may also prevent the model from distinguishing between different types of landmines. It is important to use augmentation methods that are appropriate for the real world.

7.2. Achieving balance with augmentation

While augmentation techniques can enrich a dataset and improve model performance, over-reliance on them can intuitively harm model performance. Excessive or inappropriate augmentation can cause the model to prioritize irrelevant features. Regular performance evaluation is crucial for monitoring and adjusting augmentation strategies.

7.3. Ensure consistency of the dataset across classes

Some augmentation methods may disproportionately affect different classes in the dataset. This can lead to an unbalanced dataset where some classes are overrepresented (overfitted). It is very important to use augmentation methods that maintain a consistent representation of all classes.

8.Conclusions and next steps: The role of augmentation in landmine detection

Given the limited amount of data and the dangers of experimenting with explosive objects, augmentation provides important information to improve the quality of effective landmine detection models and expand the capabilities.

Our research efforts clearly emphasize the effectiveness of data augmentation in enhancing the capabilities of the landmine detection model. Incorporating techniques such as Mix-up, Grayscale, among others, has enriched the datasets, encapsulating an expansive gamut of landmine detection scenarios. This enrichment has subsequently rendered the models more adaptable for diverse deployments.

Harnessing the YOLOv5 [14] and YOLOv8 [15] frameworks has proffered profound insights, particularly elucidating the interplay between augmentation and detection precision. However, the use of augmentation for detecting landmines requires further development. We strive for innovative augmentation methodologies, potentially using state-of-the-art models, GANs, and real-time data emulation. Nonetheless, armed with our current understanding, we are poised for further model optimization. In parallel, a mobile application project is being developed to expand the data set and classes of landmines to be recognized.

A paramount forthcoming endeavor involves subjecting the models to rigorous testing in genuine conditions. The goal is to ascertain their competency across varied topographies and ambient conditions, transitioning from the confines of labs to on-ground implementations.

It is also planned to conduct a series of experiments to improve the model's response, as this indicator is of great importance in the case of searching for explosive objects.

In conclusion, notwithstanding the substantial journey ahead, the steadfast commitment is evident: progressing towards outcomes that promise enhanced safety and preservation of human lives on a global scale. Saving lives is the cornerstone of the project, which gives us the strength to move forward with the implementation of the system for the future safe environment and happy life of future generations.

References

1. Marchuk O. How Oleksandrivka resisted the occupation.Available:https://ukrainer.net/oleksandrivka-opir/.
2. Baur J., Steinberg G., Nikulin A., Chiu K., de Smet T.S. Applying deep learning to automate UAV-based detection of scatterable landmines (2020) *Remote Sensing*, 12 (5), art. no. 859. Available:https://doi.org/10.3390/rs12050859.
3. Xiong, Z.; Zhang, X.; Hu, Q.; Han, H. IFormerFusion: Cross-Domain Frequency Information Learning for Infrared and Visible Image Fusion Based on the Inception Transformer. *Remote Sens.* 2023, *15*, 1352. Available: https://doi.org/10.3390/rs15051352.
4. Kunichik O., Tereshchnko V. Analysis of modern methods of search and classification of explosive objects. «Artificial Intelligence and Intelligent Systems», 2022. Available: https://doi.org/10.15407/jai2022.02.052.
5. Suorong Yang, Weikang Xiao, Mengcheng Zhang, Suhan Guo, Jian Zhao, Furao Shen. Image Data Augmentation for Deep Learning: A Survey. Available: https://doi.org/10.48550/arXiv.2204.08610.
6. Jung, A.B.; Wada, K.; Crall, J.; Tanaka, S.; Graving, J.; Yadav, S.; Banerjee, J.; Vecsei, G.; Kraft, A.; Borovec, J.; et al. Imgaug. 2019. Available: https://github.com/aleju/imgaug.
7. Data augmentation. TensorFlow Developers. (2023). TensorFlow (v2.14.0-rc0). Zenodo. Available: https://doi.org/10.5281/zenodo.8256979.
8. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and Flexible Image Augmentations. *Information* 2020, *11*, 125. Available: https://doi.org/10.3390/info11020125.
9. Sangdoo Yun; Dongyoon Han; Seong Joon Oh; Sanghyuk Chun; Junsuk Choe; Youngjoon Yoo; CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. Available: https://doi.org/10.48550/arXiv.1905.04899.

10. Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, David Lopez-Paz. mixup: Beyond Empirical Risk Minimization.
Available: https://doi.org/10.48550/arXiv.1710.09412.

11. Christopher Bowles, Liang Chen, Ricardo Guerrero, Paul Bentley, Roger Gunn, Alexander Hammers, David Alexander Dickie, Maria Valdés Hernández, Joanna Wardlaw, Daniel Rueckert. GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks. Available: https://doi.org/10.48550/arXiv.1810.10863.

12. Liqian Ma, Jiaojiao Meng, Shuntao Liu, Weihang Chen, Jing Xu, Rui Chen. Sim2Real2: Actively Building Explicit Physics Model for Precise Articulated Object Manipulation. Available: https://doi.org/10.48550/arXiv.2302.10693.

13. Dwyer, B., Nelson, J. (2022), Solawetz, J., et. al. Roboflow (Version 1.0) [Software].
Available: https://roboflow.com. computer vision.

14. Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, Kalen Michael, TaoXie, Jiacong Fang, imyhxy, Lorna, (Zeng Yifu), Colin Wong, Abhiram V, Diego Montes, Zhiqiang Wang, Cristi Fati, Jebastin Nadar, Laughing, … Mrinal Jain. (2022). ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation (v7.0). Zenodo.
Available: https://doi.org/10.5281/zenodo.7347926

15. Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLO by Ultralytics (Version 8.0.0) [Computer software].
Available: https://github.com/ultralytics/ultralytics.

16. O. Kunichik, Findmine_filtered Computer Vision Project,
Available:https://universe.roboflow.com/oleksandr-kunichik-sugbr/findmine_filtered.

The article has been sent to the editors 12.09.23.
After processing 13.09.23.
Submitted for printing 15.09.23.