

Early Detection of Sexually Transmitted Infections Using YOLO 12: A Deep Learning Approach

Martin Chileshe¹, Mayumbo Nyirenda¹, John Kaoma²

¹Department of Computer Science, The University of Zambia, Lusaka, Zambia

²Department of Clinical Care, Levy University Teaching Hospital, Lusaka, Zambia

Email: chilehemartin@gmail.com, mayumbo@gmail.com, chibangujkaoma111@gmail.com

How to cite this paper: Chileshe, M., Nyirenda, M. and Kaoma, J. (2025) Early Detection of Sexually Transmitted Infections Using YOLO 12: A Deep Learning Approach. *Open Journal of Applied Sciences*, 15, 1126-1144.

<https://doi.org/10.4236/ojapps.2025.154078>

Received: February 25, 2025

Accepted: April 24, 2025

Published: April 27, 2025

Copyright © 2025 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution-NonCommercial International License (CC BY-NC 4.0).

<http://creativecommons.org/licenses/by-nc/4.0/>



Open Access

Abstract

This paper focuses on the use of YOLOv12 for the early detection of Sexually Transmitted Infections, which are a global public health challenge. YOLOv12 is a deep-learning model released on February 18th, 2025. Its release has shifted from the traditional CNN-based approaches to attention-centric architecture yet still maintains high accuracy, fast inference and robust object detection capabilities with better global context modeling. This raises many interesting questions, such as whether it can perform better on real-world problems such as early detection STIs. Can the model show consistent results on different skin tones? Can it help reduce the risk of long-term effects of untreated STIs? Can YOLOv12 outperform YOLOv10 and YOLOv11? How can we validate the results? This study will answer these questions and show us how we arrived at our conclusions.

Keywords

Sexually Transmitted Infections (STIs), YOLO 12, Skin Diagnosis, Computer Vision, Generative AI

1. Introduction

Sexually transmitted infections are the world's most spread infections, especially between the ages of 15 to 50. Over 1 million curable STIs are transmitted every year, with almost a million cases per day [1]. In 2020 alone, the World Health Organisation (WHO) estimated 374 million new infections with 1 of 4 STIs: chlamydia (129 million), gonorrhoea (82 million), syphilis (7.1 million) and trichomoniasis (156 million) [2] with 20,000 cases of infertility in women annually [3].

STIs can be bacterial, viral or parasitic. They can be grouped as curable and incurable. Curable STIs include syphilis, gonorrhoea, chlamydia and trichomoniasis, to name but a few. These may be caused by bacteria or parasites [4] [5]. Incurable STIs which are caused by viruses include hepatitis B, herpes simplex virus (HSV), HIV and human papillomavirus (HPV). **Table 1** summarises these details about the infections.

Table 1. List of common STIs.

Name	Cause	Curable
chlamydia	bacteria	yes
gonorrhoea	bacteria	yes
syphilis	bacteria	yes
trichomoniasis	bacteria	yes
hepatitis B,	virus	no
herpes simplex virus (HSV)	virus	no
human papillomavirus	virus	no

YOLOv12 is a group of deep learning models or AI models that were released on February 18th, 2025. **Figure 1** shows a list of YOLOv12 models along with their mAp, Speed and Number of Parameters. These models are currently maintained by Ultralytics, which has made it much easier to configure and run experiments faster.

Model	size (pixels)	mAP ^{val} ₅₀₋₉₅	Speed CPU ONNX (ms)	Speed T4 TensorRT (ms)	params (M)	FLOPs (B)
YOLO12n	640	40.6	-	1.64	2.6	6.5
YOLO12s	640	48.0	-	2.61	9.3	21.4
YOLO12m	640	52.5	-	4.86	20.2	67.5
YOLO12l	640	53.7	-	6.77	26.4	88.9
YOLO12x	640	55.2	-	11.79	59.1	199.0

Figure 1. A list of YOLOv12 models.

2. Methodology

The first step in our series of experiments is to identify infections that show visible symptoms as early as one to three days after exposure to the bacteria or virus. We shall collect this information from various research journals, books, medical websites, blogs and the internet. Once this information is gathered, we will proceed to data collection, which will provide us with thousands of images displaying symptoms of these infections. Finally, we will proceed to the part where the experiments are performed by training a YOLOv12 model on the collected data [6].

2.1. Data Collection

In deep learning experiments, data collection is the most crucial step because the model largely depends on the quantity and quality of the collected data [7]. The model can only be as good as the information it's trained with. If we use a small dataset, we end up with a biased model that can only perform well on the data it has seen before, a situation called underfitting [8] [9].

It is also important to capture data encompassing as many variations as possible otherwise, we end up with a model that can only perform well on paper or in the lab but is useless in the real world. This is referred to as an overfit model [10]. Considering these factors, we will use a combination of data collection techniques, focusing only on infections that show visual symptoms. **Table 2** shows a list of common STIs with their symptoms.

Table 2. List of common STIs with their symptoms.

Name	Cause	Early Visual Symptoms
chlamydia	bacteria	
gonorrhoea	bacteria	
syphilis	bacteria	
trichomoniasis	parasites	Genital redness or swelling
hepatitis B	virus	Yellowing of the skin and whites of the eyes (jaundice)
herpes simplex virus (HSV)	virus	Sores or blisters around the mouth or genitals
human papillomavirus	virus	

2.2. Data Sources

Data was collected from the following four major sources since not one source could provide adequate data to enable us to train a model effectively.

Kaggle

Kaggle is an online community of data scientists, machine learning experts and researchers. It was created to boost development in AI. Kaggle hosts thousands of projects through international competitions. The data and projects hosted are free and open source [11].

CDC: Centers for Disease Control and Prevention

The Centers for Disease Control and Prevention is the national public health agency of the United States. It is a United States federal agency under the Department of Health and Human Services and is headquartered in Atlanta, Georgia. CDC works 24/7 to protect America from health, safety and security threats, both foreign and in the U.S. [12].

DermNet

DermNet is the world's premier free dermatology resource designed for healthcare professionals. It serves as a comprehensive database of skin conditions.

Since almost everyone encounters a skin issue at some point, having a reliable, independent, and easily accessible source of information is essential for both practitioners and patients. DermNet fulfils that need. It is trustworthy, always free, and available to all anytime. See links under supporting information.

Atlas Dermatological

Atlas Dermatológico is a Spanish-language dermatology resource that provides a collection of images and information on various skin conditions. It is commonly used by healthcare professionals, medical students, and the general public to aid in the diagnosis and understanding of dermatological diseases. See links under supporting information.

2.3. Data Generation

With the advancements of transformer-based neural networks [13] [14] and Generative Adversarial Networks (GANs) [15] [16], it is now possible to generate similar data based on a given input. If we have an image of a given infected area, the transformer/GAN model can generate images containing similar content with variations given by the command. Some already existing tools that can be used to generate multimedia data include openart.ai, ChatGPT, DALL-E, DALL-E 2, and DALL-E 3. Note that DALL-E, DALL-E 2, and DALL-E 3 are models, while ChatGPT is a chatbot. These methods are now being referred to as Generative AI. We will explore some of these data generation techniques to see how good data generation can be for training a deep learning model such as YOLO 12.

2.4. Data Augmentation

Data augmentation is a series of techniques that can be applied to images to generate new versions of the given images. The methods involve using various algorithms to alter the images to resemble real-world variations [17]. Data Generation and Data augmentation both aim to produce data, but the way they do it is different. Data augmentation transforms existing data, while data generation creates new samples that replicate the original data's patterns.

There are several techniques which include: blur, flip, 90° rotate, Crop, Rotation, Shear, Grayscale, Hue, Saturation, Brightness, Exposure, Noise, cutout, Mosaic, adversarial training, geometric transformations, colour space transformations, kernel filters, mixing images, random erasing, feature space augmentation, GAN-based augmentation, neural style transfer, and meta-learning schemes [18] [19]. Not all techniques produce the desired results, so we will not use all. The experiments will be conducted, as well as the recommendations given by Alhassan Mumuni and Fuseini Mumuni [20]. Shorten & Khoshgoftaar also give good insights on the mentioned techniques.

2.5. Data Processing

Data processing involves cleaning up, preprocessing and labeling. All three 3 techniques are essential, and they will all be used. We will start with data cleaning [21].

Data Cleaning

It is important to note that data collected either from the internet or the real world is never in the desired format and may contain unwanted content [22]. For example, images that are downloaded from the internet can never have the same size; however, YOLO requires all the images to be of the same size before beginning the training. Similarly, web scraping may not collect only the content described in the query or command, so data has to be manually inserted and content with copyright issues removed [23].

Preprocessing

Preprocessing is the process of transforming data for analysis. In mathematical terms, this is applying a transformation on vectors X_{ik} to a set of new vectors Y_{ik}

$$Y_{ij} = T(X_{ik}) \quad (1)$$

- 1) In this equation: Y_{ij} preserves the “useful information” in X_{ik}
- 2) Y_{ij} eliminates at least one of the problems in X_{ik}
- 3) Y_{ij} are more useful than X_{ik} in the above relation

The overall goal of these transformations is to discover valuable information from the data [24] [25] while also eliminating outliers. Ensuring that we preserve only the features of interest allows us to reduce training time and increase performance. Some techniques include: isolating objects, Dynamic Crop, Grey Scale, Auto-Adjust Contrast, Title, Modify Classes, and Filter B Tag. For more information on this, I strongly recommend S. B. Kotsiantis, D. Kanellopoulos and P. E. Pintelas on Data Preprocessing for Supervised Learning [26].

Data Labeling

Data labelling is the final step in data processing, specifically for object detection tasks. Each image is labelled in this exercise to indicate the data points and what they represent. **Figure 2** shows a labelled image with a bus, a car and a person in it. These object locations are called data points. Here, we draw a bounding box around each data point and indicate what it represents as seen.

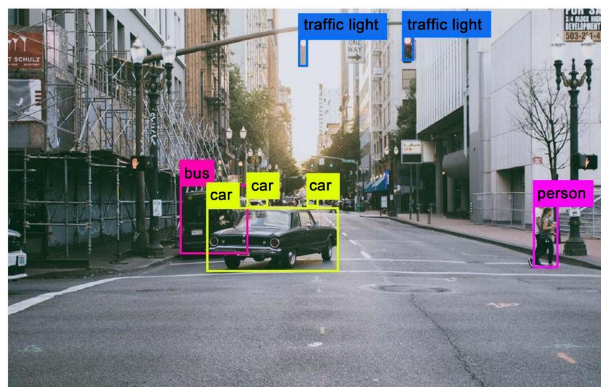


Figure 2. Data labeling.

Many tools can help us achieve this, including the Free and Open Source (FOSS) labelImg, which can be installed on any operating system. Label Studio, is

a FOSS and flexible data labeling tool for all data types. However, this process is very slow and cumbersome, but thanks to Michael Desmond, Evelyn Duesterwald, Kristina Brimijoin, Michelle Brachman, and Qian Pan, who proposed Semi-Automated Data Labeling [27], which helps speed up the processing by guiding the labeller. One outstanding tool is RoboFlow's semi-automatic labelling platform because it allows the use of a trained model for labelling. The results may not be accurate, so it still requires manual inspection.

Vector Analysis in Object Detection

Vector analysis plays a crucial role in object detection by representing objects, bounding boxes, and feature maps as vectors and performing operations on them [28]. Key applications include:

1) Feature Extraction using CNNs

a) Convolutional layers extract spatial features from images, represented as high-dimensional vectors.

b) Feature maps are processed using filters (kernels), which apply vector transformations.

2) Bounding Box Representation & Regression

a) Objects in images are enclosed in bounding boxes, represented as 4D vectors:

$$B = (x, y, w, h) \quad (2)$$

where x, y are the centre coordinates, and w, h are the width and height.

b) Models predict bounding boxes using vector regression techniques.

c) Higher IoU indicates better detection accuracy.

3) IoU (Intersection over Union) for Object Localization

a) IoU is a vector-based metric used to measure the overlap between predicted and ground-truth bounding boxes:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (3)$$

b) Higher IoU indicates better detection accuracy.

4) Anchor Boxes & Priors

a) Predefined vectorized bounding boxes (anchors) are used to detect objects of varying sizes.

b) Networks adjust these anchors to fit detected objects.

5) Non-Maximum Suppression (NMS)

a) A vector-based algorithm filters overlapping bounding boxes by keeping the one with the highest confidence score while suppressing others.

6) Object Classification Using Fully Connected Layers

a) Extracted feature vectors are passed to a classifier (e.g., softmax or sigmoid) to assign object labels.

7) Transformers & Attention in Detection (DETR)

a) Transformer-based models like DETR use self-attention mechanisms, computing weighted dot products of vectorized object representations.

To analyse our dataset, we use a scatterplot, which will visualise the relationships between the classes. We can colour our scatterplot based on class labels, the

number of objects in each image, or their train/validation/test split.

This type of plot is useful for spotting patterns and outliers in our dataset. For instance, if the train/validation/test sets are highly disjointed, they might not be representative, which could lead to model performance issues. Similarly, if a single instance of a class appears isolated, it could indicate an edge case or a potential labelling error. **Figure 3** is a scatterplot generated using our dataset.

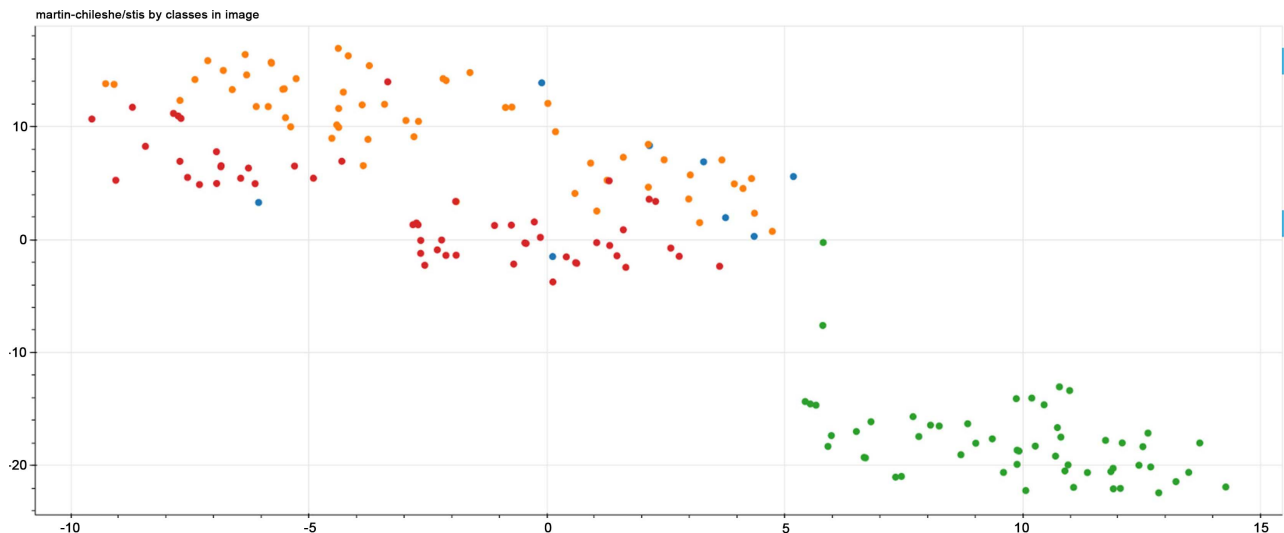


Figure 3. Scatterplot. Each colour represents a class.

2.6. Training

Model training is the final stage where we feed our data into the model and pay attention to the running performance, the number of epochs, training time and accuracy. There are a lot of parameters used for model training some include train/box_loss, train/class_loss, train/df_l_loss, metrics/precision (B), metrics/recall (B), val/box_loss, val/cls_loss, val/df_l_loss, metrics/mAP50 (B) and metrics/mAP50-95 (B). There are also hyperparameters, which are related to the architecture of the model. For an in-depth understanding of hyperparameters, Yang, L., & Shami, A. [29] give a good overview. **Figure 4** shows the graphs at the final stage of the training process.

Recall, Precision and Average Precision

These terms are commonly used in information retrieval and machine learning, particularly in evaluating classification models.

1) Recall (Sensitivity or True Positive Rate):

a) Measures the ability of a model to identify all relevant instances.

b) Formula:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \quad (4)$$

c) High recall means the model captures most of the relevant cases but may include many false positives.

2) Precision (Positive Predictive Value):

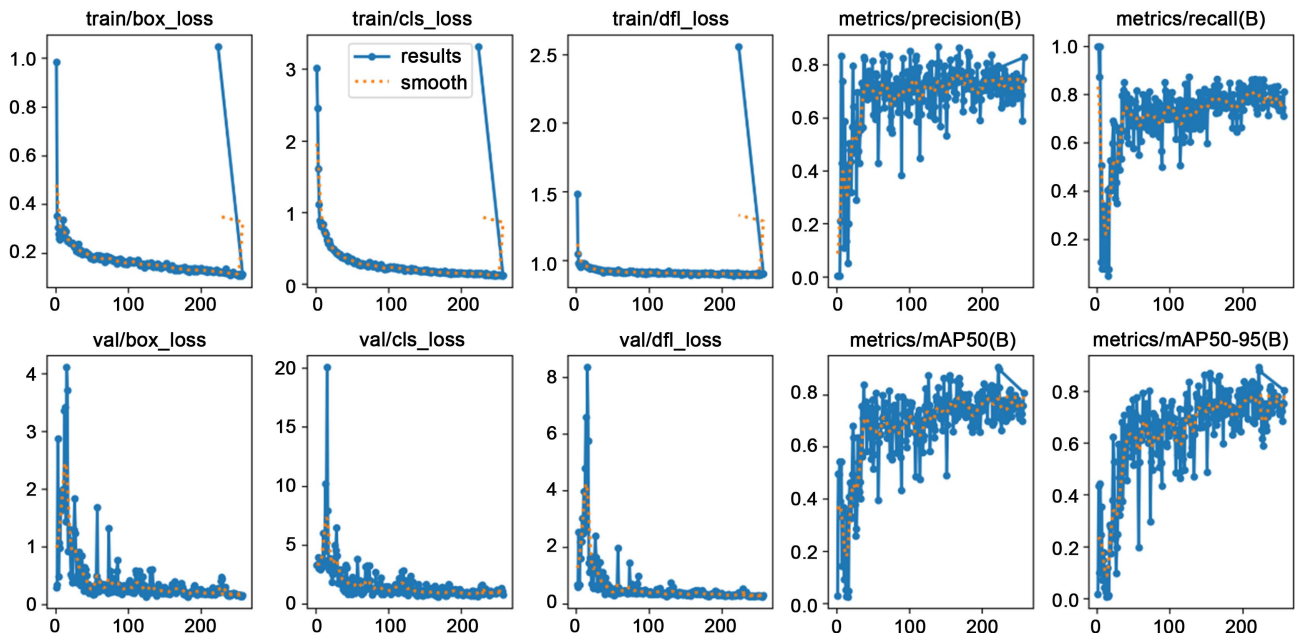


Figure 4. Training graphs.

a) Measures how many of the predicted positive instances are correct.

b) Formula:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \quad (5)$$

c) High precision means that most of the predicted positives are correct, but the model may miss some relevant cases.

3) Average Precision (AP):

a) Measures the overall performance of a model across different recall levels.

b) It is the area under the Precision-Recall (PR) curve, computed as:

$$\text{AP} = \sum (R_n - R_{n-1}) P_n \quad (6)$$

where R_n and R_{n-1} are recall values at different thresholds, and P_n is the corresponding precision [30] [31].

c) It is often used in object detection and ranking problems.

Figure 5 shows the final Recall, Precision and Average Precision scores that we obtained from our training.

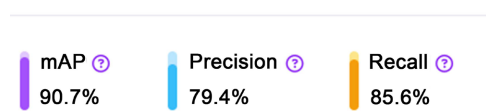


Figure 5. Recall, precision and average precision.

3. Results, Data Transformations Used and Validation

In this section, we review the results obtained from training our models, starting with a comparative analysis of how various models performed on the same dataset, fine-tuning YOLOv12-S and finally discussing validation strategies.

3.1. Results

A Comparative Analysis

YOLOv12 comes in five different sizes (N, S, M, L, and X) with parameters ranging from 2.6 M to 59.1 M, striking an ideal balance between accuracy and speed. Compared to YOLOv10-S and YOLOv11-S, YOLOv12s did not show significant differences with the Precision-Recall, as seen in **Figures 6-9**. To train a dataset of 776 images, YOLOv12-S took 0.683 hours, while YOLOv11-S took 0.410 hours, and YOLOv10-S took 0.471 hours on a T4 GPU.

As shown in **Figures 10-13**, all models performed very well on Hepatitis B, where YOLOv11 and 10 show a score of 1.00 while YOLOv12 scores 0.96; however, YOLOv11 and YOLOv10 struggled with generalization as they scored very low on herpes simplex but extremely high on hepatitis B and syphilis. YOLOv12, on the other hand, scored consistent results across all classes despite the scores being low.

Fine-Tuning

Figure 14 shows the final results obtained from fine-tuning YOLOv12-S, which was trained on a data set of 1500 images. The training was done on a T4 GPU,

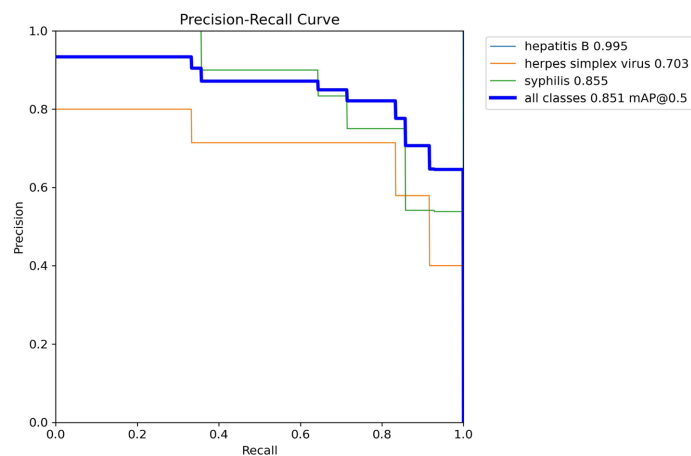


Figure 6. YOLOv10 PR curve.

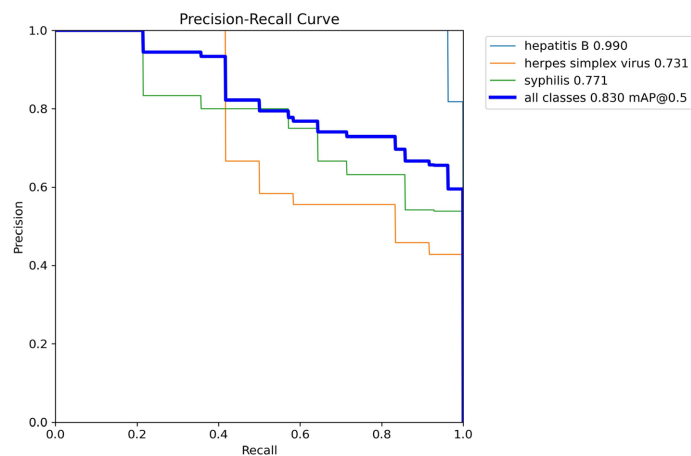


Figure 7. YOLOv12 PR curve.

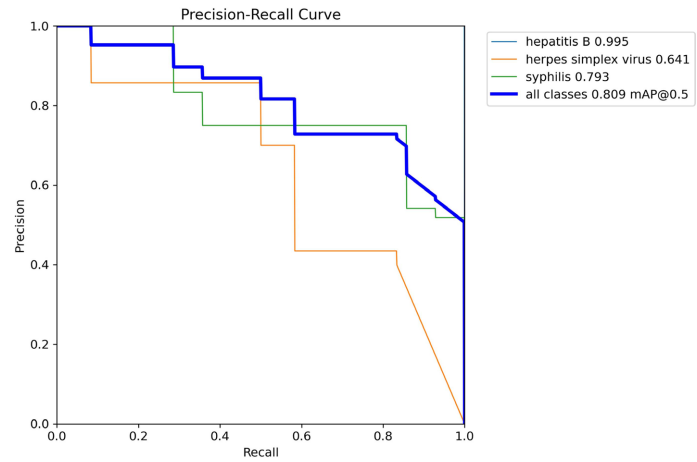


Figure 8. YOLOv11 PR curve.

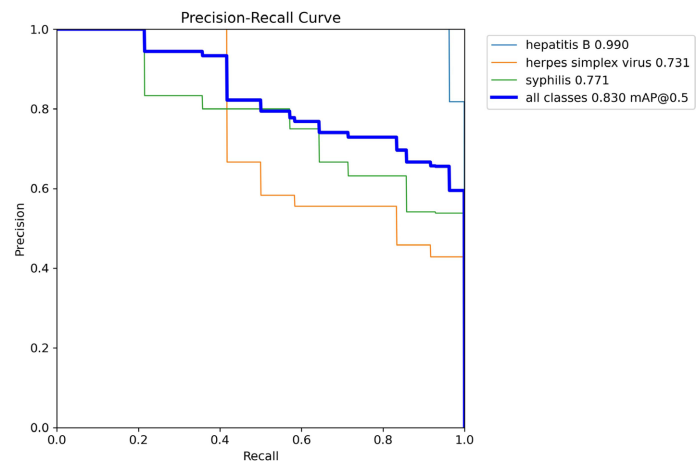


Figure 9. YOLOv12 PR curve.



Figure 10. YOLOv10 confusion matrix.

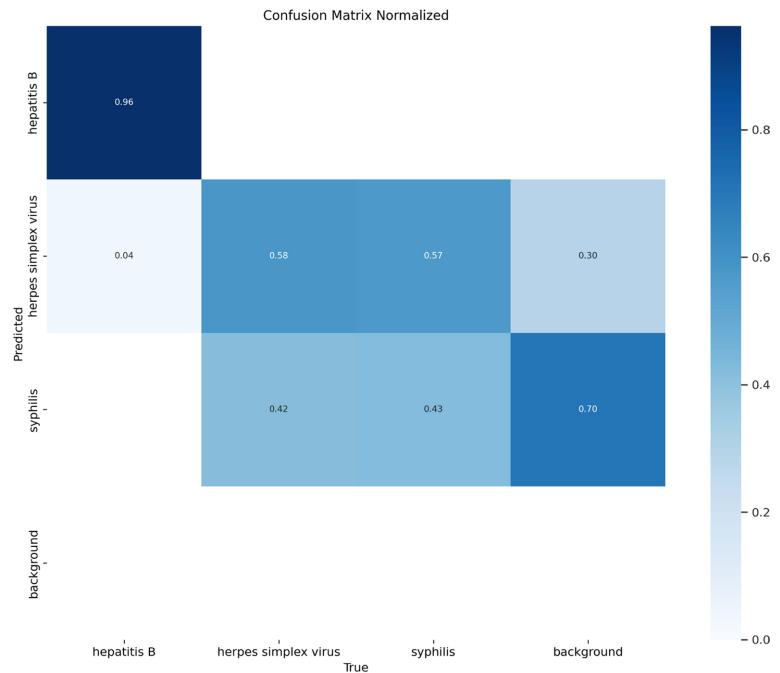


Figure 11. YOLOv12 confusion matrix.

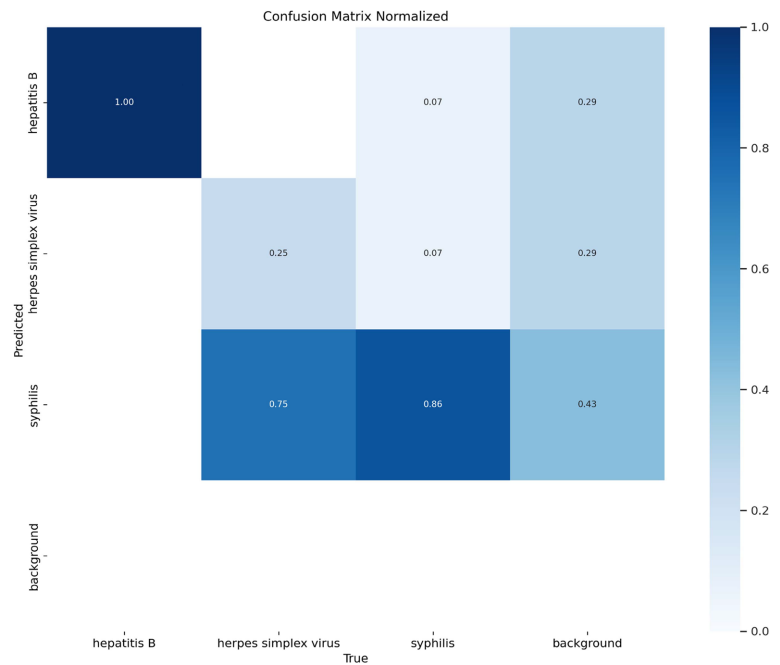


Figure 12. YOLOv11 confusion matrix.

which ran for 150 epochs, and it took 1.32 hours to complete. Image size was maintained at 640 × 640.

Figures 15-20 show evaluations on actual unseen data.

These results show substantial advancement in attention-based real-time object identification by matching or exceeding state-of-the-art accuracy without sacrificing detection speed.

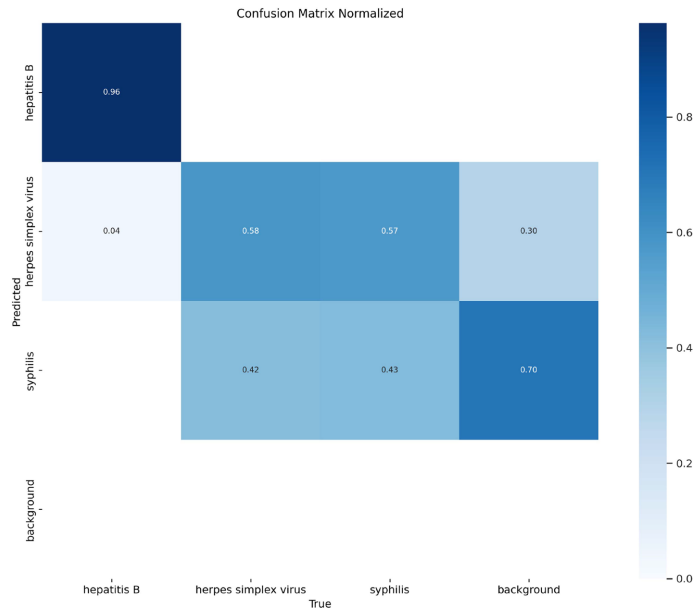


Figure 13. YOLOv12 confusion matrix.



Figure 14. Average precision by class (mAP50).



Figure 15. Herpes simplex 98% accurate.



Figure 16. Herpes simplex 99% accurate.



Figure 17. Hepatitis B is 93% accurate.



Figure 18. Syphilis is 84% accurate.



Figure 19. Syphilis is 98% accurate.

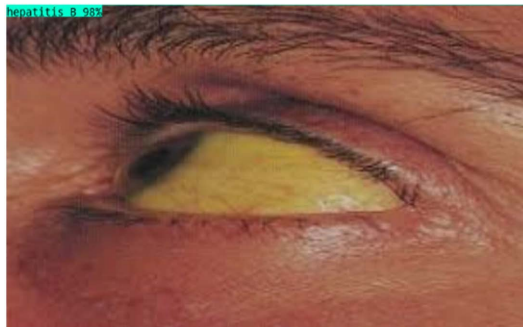


Figure 20. Hepatitis B is 98% accurate.

3.2. Data Transformations Used

Preprocessing

Preprocessing was introduced earlier in chapter 2.2.2. In our dataset, we applied a total of 4 preprocessing algorithms.

1) Auto-Orient

Images have metadata that indicates the orientation of each image and how it should be displayed on screens. When pictures are taken using a camera, images are stored with the same value for the orientation metadata, regardless of whether the camera is in landscape or portrait. Due to this, training images must be randomly reoriented to allow the model flexibility for objects and bounding boxes in different orientations. Roboflow has a feature called auto-orient, which automatically changes the orientation of the images [32].

2) Isolate Objects

In object detection tasks, isolating an object refers to extracting or segmenting a detected object from its surroundings. This can involve:

- Bounding Box Extraction—Drawing a box around the detected object to separate it from the rest of the image.
- Instance Segmentation—Identifying the exact shape and boundaries of the object rather than just a box.
- Masking—Creating a binary or soft mask to remove the background and retain only the object of interest.
- Cropping—Cutting out the detected object from the image for further processing or analysis.

3) Resize

YOLOv12 requires all images to be of the same size, which is 640×640 . Roboflow offers several resize algorithms, including stretch to with (centres crop) in, fit within, fit (reflect edges) in, fit (black edges) in, and fit (white edges) in. In our case, we used stretch to 640×640 . For a detailed explanation of these different algorithms, please read the RoboFlow blog on resizing your images [33].

4) Filter Null

Filter null allows a few images without any of the desired objects to be added to the dataset so that the model can learn that not all images have objects. The number of such images should not be too many. For our dataset, we used 57%. For more on filter null, read this RoboFlow blog on Manage Classes

Augmentations

Data augmentation was introduced in section 2.1.3. For our dataset, we used the following techniques, which gave us these results.

1) Flip

Flipping images horizontally or vertically can help make a model insensitive to subject orientations. Both vertical and horizontal flips were applied [34].

2) Shear

In the context of object detection and image processing, shear refers to a geometric transformation that distorts an image along a particular axis, shifting parts

of the image in one direction while keeping the other axis fixed. This transformation is commonly used in data augmentation to make models more robust to variations in object appearance.

For example:

- Shear in the X-axis tilts the image left or right.
- Shear in the Y-axis stretches the image up or down.

Shearing is particularly useful in deep learning to help models generalize better by simulating real-world distortions, such as perspective changes. We applied a shear of 10% vertical and 10% horizontal.

3) Blur

In object detection and image processing, blur refers to a technique that reduces sharpness and detail in an image by averaging pixel values. This can be used for:

Types of Blur

- Gaussian Blur—Applies a Gaussian function to smooth the image, reducing noise while maintaining edges.
- Motion Blur—Simulates movement by smearing pixels in a specific direction.
- Median Blur—Replaces each pixel with the median value of its neighbours, which is useful for removing salt-and-pepper noise.
- BoxBlur—Averages surrounding pixels uniformly, creating a simple blur effect.
- Bilateral Blur—Smooths while preserving edges, useful in tasks like edge detection preprocessing.

Uses in Object Detection

- Data Augmentation: Blurring can help models generalize by making them robust to low-quality images.
- Preprocessing: Helps remove noise before edge detection (e.g., Canny edge detection).
- Privacy Protection: Used to obscure sensitive details in images.

Our dataset uses a random Gaussian blur of 2.5 px. Read more on random blur [35].

3.3. Validation

To validate our model, we split the data into training, validation, and testing sets with percentages indicated in **Figure 21**.

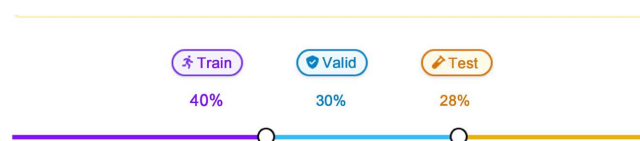


Figure 21. Data split.

The purpose of this validation during training is to examine how well the model performs on unseen data, thereby adjusting parameters and hyperparameters to avoid biases, overfitting or underfitting [36].

Clinical Diagnosis

Clinical Diagnosis is the process of identifying a health condition, injury, or disease. It's based on a patient's symptoms, medical history, and physical exam. In addition to the validation performed during the training process, follow-up questions will have to be asked of a patient, which target symptoms related to the model's prediction of the image submitted. For instance, if the model is given an image containing a syphilis ulcer and it is more than 70% confident that there is an ulcer in the image, the mobile app then queries the database for other symptoms related to syphilis such as Swollen lymph glands in the groin or neck, Fever, Patchy hair loss, Muscle and joint aches, Headaches and Tiredness [37]. Once the patient provides positive results showing similarities, the app will be more confident in telling the patient what it thinks the patient has been infected with.

Medical Personnel

In the end, medical personnel have been engaged on the possibility of performing lab [38]. We do not currently have results on this but will provide more information in the next article. In the meantime, the choice is up to the patient to rely solely on the results provided by the mobile app or engage medical personnel. A user consent agreement has been included in the app for this purpose. The mobile app provides the option for them to contact registered and qualified medical personnel for additional validation.

4. Discussion

As seen by the results, YOLOv12 can achieve high accuracy on STIs. To answer the questions we had before the experiments:

1) Will the model be able to perform better on real-world problems, such as the early detection of STIs?

Based on our results shown under the results section and a test set of 420 image samples of the 3 infections, the model accurately predicted the correct labels of 383 images from the test set. So, yes, the model does perform well in the real world.

2) Can the model show consistent results on different skin tones?

As indicated by the test images in **Figures 15-20**, we note that there are different skin tones, and the model does predict the infections correctly. So yes. The model does perform well on different skin ton.

3) Can it help reduce the risk of long-term effects of untreated STIs?

Since the model does provide quick diagnosis, especially in situations where health professionals are not immediately available, this can be a powerful tool to help prevent long-term effects, especially with enough sensitization.

4) Can YOLOv12 outperform YOLOv10 and YOLOv11?

YOLOv12 outperforms the other two models when we compare the performance holistically, overall performance on multiple classes.

5) How can we validate the results?

As indicated in the validation section, we do note that there are several techniques to verify the results of our model predictions. They include a validation set

of 450 images and 420 test images. In addition, follow-up questions compare other symptoms not detectable by our model to the results obtained from the model predictions. This boosts the confidence in the model predictions.

5. Conclusion

From this research, we have confirmed that YOLOv12 supersedes the earlier models in terms of generalisation since the other models showed great variations in the scores for different classes. Having noted that YOLOv12 was doing a better job across the 3 classes, it went further and fine-tuned our model by training it on a larger database with 150 epochs, which indicates that, indeed, YOLOv12 was the better model for global context modelling and better suited for analyzing medical images consisting of symptoms of sexually transmitted diseases. We have also concluded that of all the known sexually transmitted diseases, only hepatitis B, Herpes Simplex and Syphilis can be correctly predicted because they show visual characteristics that an AI model can learn, YOLOv12.

We have also indicated how these results were and can be validated further. To make this more robust, further treatment can be done with more data.

Finally, we have also presented several techniques for data collection, including web scraping, data cleaning, scatter plots, vector analysis, data generation and data augmentations. In the real world, where data is unavailable in large quantities, you must use some of these proven methods for a better conclusion.

Supporting Information

S1 Link. Skin Infections (Website) Skin infections on Kaggle.

S2 Link. Syphilis Images (Website) Some Syphilis images from CDC.

S3 Link. Skin images infections (Repository) Demnet images.

S4 Link. Skin images atlas (Atlas) Atlas Dermatológico.

Acknowledgements

We acknowledge the support and guidance from the co-authors for all the support rendered. We also acknowledge Kaggle, CDC, Dermnet and Atlas Dermatológico for making their datasets available to the public.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] World Health Organization (2018) WHO Expert Consultation on Rabies: Third Report (Vol. 1012). World Health Organization.
- [2] World Health Organization (2018) Sexually Transmitted Infections (STIs). World Health Organization.
- [3] Garcia, M.R., Leslie, S.W. and Wray, A.A. (2024) Sexually Transmitted Infections. National Library of Medicine.

- [4] Igietseme, J.U., Omosun, Y. and Black, C.M. (2015) Bacterial Sexually Transmitted Infections (STIs). *Molecular Medical Microbiology*, **3**, 1403-1420. <https://doi.org/10.1016/b978-0-12-397169-2.00078-0>
- [5] Whitlow, C.B. (2004) Bacterial Sexually Transmitted Diseases. National Library of Medicine.
- [6] Tian, Y.J., Ye, Q.X. and Doermann, D. (2025) YOLOv12: Attention-Centric Real-Time Object Detectors. arXiv:2502.12524.
- [7] Whang, S.E. and Lee, J. (2020) Data Collection and Quality Challenges for Deep Learning. *Proceedings of the VLDB Endowment*, **13**, 3429-3432. <https://doi.org/10.14778/3415478.3415562>
- [8] Aliferis, C. and Simon, G. (2024) Overfitting, Underfitting and General Model Overconfidence and Under-Performance Pitfalls and Best Practices in Machine Learning and AI. In: Simon, G.J. and Aliferis, C., Eds., *Artificial Intelligence and Machine Learning in Health Care and Medical Sciences*, Springer, 477-524. https://doi.org/10.1007/978-3-031-39355-6_10
- [9] Sehra, S., Flores, D. and Montanez, G.D. (2021) Undecidability of Underfitting in Learning Algorithms. 2021 *2nd International Conference on Computing and Data Science (CDS)*, Stanford, 28-29 January 2021, 591-594. <https://doi.org/10.1109/cds52072.2021.00107>
- [10] Ying, X. (2019) An Overview of Overfitting and Its Solutions. *Journal of Physics: Conference Series*, **1168**, Article ID: 022022. <https://doi.org/10.1088/1742-6596/1168/2/022022>
- [11] (2010) Students, Seasoned Professionals, and Distinguished Researchers. <https://www.kaggle.com/datasets>
- [12] Mountin, J.W. (1946) CDC: Centers for Disease Control and Prevention. <https://www.cdc.gov/sti/php/training/picture-cards.html>
- [13] Islam, S., Elmekki, H., Elsebai, A., Bentahar, J., Drawel, N., Rjoub, G., *et al.* (2024) A Comprehensive Survey on Applications of Transformers for Deep Learning Tasks. *Expert Systems with Applications*, **241**, Article ID: 122666. <https://doi.org/10.1016/j.eswa.2023.122666>
- [14] Lu, Y., Shen, M., Wang, H., Wang, X., van Rechem, C., Fu, T. and Wei, W. (2023) Machine Learning for Synthetic Data Generation: A Review. arXiv: 2302.04062.
- [15] Saxena, D. and Cao, J.N. (2020) Generative Adversarial Networks (GANs): Challenges, Solutions, and Future Directions. arXiv: 2005.00065.
- [16] Pan, Z.Q., Yu, W.J., Yi, X.K., Khan, A., Yuan, F. and Zheng, Y.H. (2019) Recent Progress on Generative Adversarial Networks (GANs): A Survey. *IEEE Access*, **7**, 36322-36333.
- [17] van Dyk, D.A. and Meng, X. (2001) The Art of Data Augmentation. *Journal of Computational and Graphical Statistics*, **10**, 1-50. <https://doi.org/10.1198/10618600152418584>
- [18] Shorten, C. and Khoshgoftaar, T.M. (2019) A Survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, **6**, Article No. 60. <https://doi.org/10.1186/s40537-019-0197-0>
- [19] Maharana, K., Mondal, S. and Nemade, B. (2022) A Review: Data Pre-Processing and Data Augmentation Techniques. *Global Transitions Proceedings*, **3**, 91-99. <https://doi.org/10.1016/j.gltp.2022.04.020>
- [20] Mumuni, A. and Mumuni, F. (2022) Data Augmentation: A Comprehensive Survey of Modern Approaches. *Array*, **16**, Article ID: 100258. <https://doi.org/10.1016/j.array.2022.100258>

- [21] Wickham, H. (2016) Data Analysis. In: Wickham, H., Ed., *ggplot2*, Springer, 189-201. https://doi.org/10.1007/978-3-319-24277-4_9
- [22] Chu, X., Ilyas, I.F., Krishnan, S. and Wang, J. (2016) Data Cleaning: Overview and Emerging Challenges. *Proceedings of the 2016 International Conference on Management of Data*, San Francisco, 26 June-1 July 2016, 2201-2206. <https://doi.org/10.1145/2882903.2912574>
- [23] Rahm, E. and Do, H.H. (2000) Data Cleaning: Problems and Current Approaches. *IEEE Data Engineering Bulletin*, **23**, 3-13.
- [24] Famili, A., Shen, W., Weber, R. and Simoudis, E. (1997) Data Preprocessing and Intelligent Data Analysis. *Intelligent Data Analysis*, **1**, 3-23. <https://doi.org/10.3233/jida-1997-1102>
- [25] Alasadi, S.A. and Bhaya, W.S. (2017) Review of Data Preprocessing Techniques in Data Mining. *Journal of Engineering and Applied Sciences*, **12**, 4102-4107.
- [26] Kotsiantis, S.B., Kanellopoulos, D. and Pintelas, P.E. (2006) Data Preprocessing for Supervised Learning. *International Journal of Computer Science*, **1**, 111-117.
- [27] Desmond, M., Duesterwald, E., Brimijoin, K., Brachman, M. and Pan, Q. (2021) Semi-Automated Data Labeling. *NeurIPS 2020 Competition and Demonstration Track*, 6-12 December 2020, 156-169.
- [28] Newell, H.E. (2006) Vector Analysis. Courier Corporation.
- [29] Yang, L. and Shami, A. (2020) On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice. *Neurocomputing*, **415**, 295-316. <https://doi.org/10.1016/j.neucom.2020.07.061>
- [30] Zhu, M. (2004) Recall, Precision and Average Precision. Department of Statistics and Actuarial Science, University of Waterloo, 2(30), 6.
- [31] Robertson, S. (2008) A New Interpretation of Average Precision. *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Singapore, 20-24 July 2008, 689-690. <https://doi.org/10.1145/1390334.1390453>
- [32] Dwyer, B. (2020) When Should I Auto-Orient My Images? Roboflow Blog. <https://blog.roboflow.com/exif-auto-orientation/>
- [33] Nelson, J. (2020) You Might Be Resizing Your Images Incorrectly. Roboflow Blog. <https://blog.roboflow.com/you-might-be-resizing-your-images-incorrectly/>
- [34] Nelson, J. (2020) How Flip Augmentation Improves Model Performance. Roboflow Blog. <https://blog.roboflow.com/how-flip-augmentation-improves-model-performance/>
- [35] Nelson, J. (2020) The Importance of Blur as an Image Augmentation Technique. Roboflow Blog. <https://blog.roboflow.com/using-blur-in-computer-vision-preprocessing/>
- [36] Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., et al. (2023) Hyperparameter Optimization: Foundations, Algorithms, Best Practices, and Open Challenges. *WIREs Data Mining and Knowledge Discovery*, **13**, e1484. <https://doi.org/10.1002/widm.1484>
- [37] Brown, D.L. and Frank, J.E. (2003) Diagnosis and Management of Syphilis. *American Family Physician*, **68**, 283-290.
- [38] Zerr, I. and Poser, S. (2002) Clinical Diagnosis and Differential Diagnosis of CJD and vCJD. *APMIS*, **110**, 88-98. <https://doi.org/10.1034/j.1600-0463.2002.100111.x>