UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

VICTORIA PANTOJA DO AMARAL

Enhancing mosquito egg counting accuracy through deep learning image restoration

Recife
2025

VICTORIA PANTOJA DO AMARAL

Enhancing Mosquito Egg Counting Accuracy through Deep Learning Image Restoration

Trabalho apresentado ao Programa de Graduação em Sistemas de Informação da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Graduado em Sistemas de Informação.

Orientador: Leandro Maciel Almeida

Recife

2025

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Amaral, Victoria Pantoja do.
Enhancing mosquito egg counting accuracy through deep learning image restoration / Victoria Pantoja do Amaral. - Recife, 2025.
42 : il., tab.

Orientador(a): Leandro Maciel Almeida
Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Informática, Sistemas de Informação - Bacharelado, 2025.
Inclui referências, apêndices, anexos.

1. mosquito surveillance. 2. deep learning. 3. image restoration. 4. aedes aegypti. I. Almeida, Leandro Maciel. (Orientação). II. Título.

000 CDD (22.ed.)

VICTORIA PANTOJA DO AMARAL

ENHANCING MOSQUITO EGG COUNTING ACCURACY THROUGH DEEP LEARNING IMAGE RESTORATION

Trabalho de Conclusão de Curso apresentado ao Programa de Graduação em Sistemas de Informação da Universidade Federal de Pernambuco como requisito para a obtenção do título de Graduado em Sistemas de Informação.

Aprovado em: 01/04/2025.

BANCA EXAMINADORA

Prof. Dr. Leandro Maciel Almeida (Orientador)
Universidade Federal de Pernambuco - UFPE

Prof. Dr. Fernando Maciano de Paula Neto
Universidade Federal de Pernambuco - UFPE

ABSTRACT

Monitoring Aedes aegypti populations is crucial for dengue prevention, with egg counts collected from ovitraps serving as a primary method for tracking. This study addresses the limitations of smartphone-captured images, which may suffer from motion blur, defocus, and noise—factors that significantly impair automated counting accuracy. We evaluated three deep learning image restoration models—MPRNet, Real-ESRGAN, and Restormer—to enhance image quality prior to automated egg detection. Using a dataset of 82 ovitrap images, the models were trained and evaluated based on both perceptual metrics (PSNR and NIQE) and their impact on automated egg counting compared to manual counts. Among the tested models, Real-ESRGAN achieved the best performance, improving counting accuracy from 78.4% to 106.5%. In contrast, MPRNet and Restormer performed poorly with the provided training data, reaching 331.7% and 1.5% accuracy, respectively. The results demonstrate that appropriate image enhancement techniques can improve the precision of mosquito egg counting under real-world conditions without requiring specialized equipment, potentially contributing to more efficient disease prevention strategies.

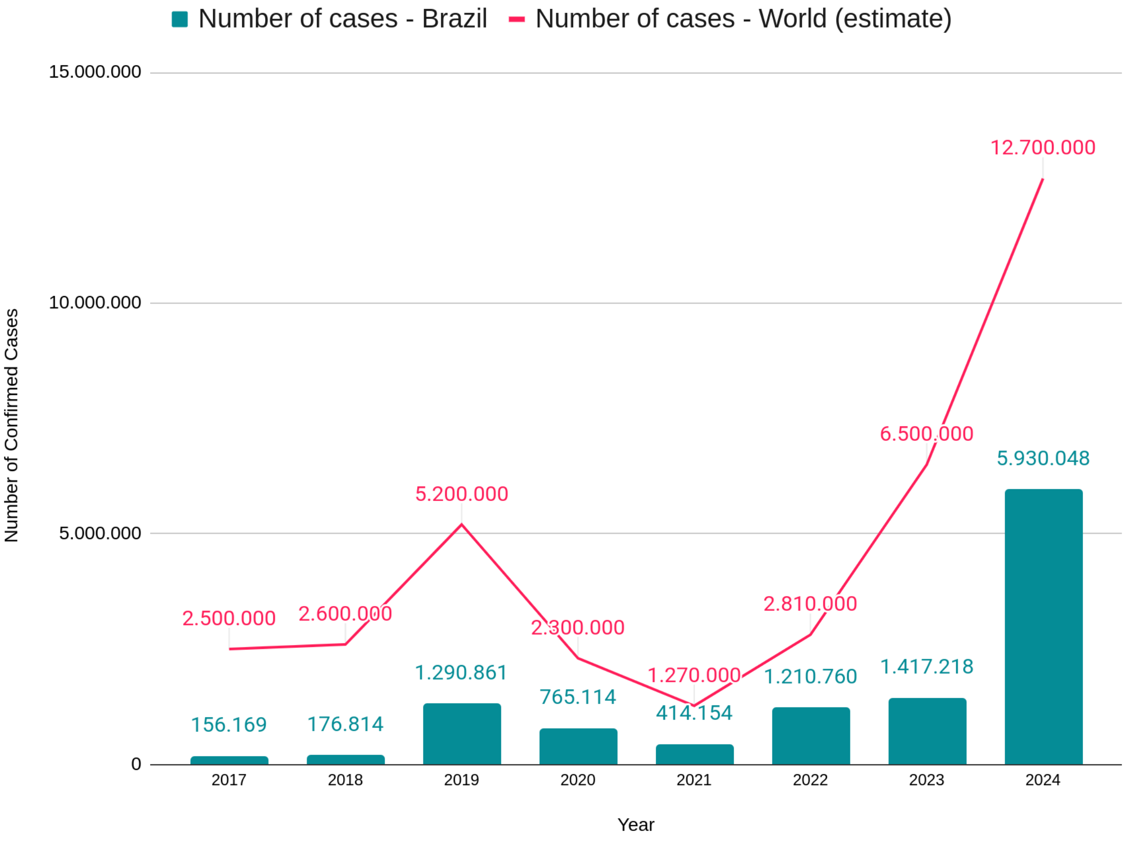
Keywords: mosquito surveillance; deep learning; image restoration; aedes aegypti.

SUMMARY

1. INTRODUCTION

Aedes *aegypti* mosquitoes are primary vectors for significant diseases such as chikungunya, yellow fever, Zika, and dengue, predominantly affecting tropical and subtropical regions, especially in the Americas. Since 2019 the number of dengue fever cases has rapidly increased. That year alone saw over 5.2 million reported cases worldwide, marking a significant increase compared to previous years. By April 2024, the World Health Organization had already documented over 7.6 million cases, surpassing the total of 4.6 million cases reported in 2023 (WORLD HEALTH ORGANIZATION, 2014). This upward trend imposes a substantial economic burden, with estimates indicating that the annual cost of dengue in Latin America exceeds US$3 billion (LASERNA et al, 2018).

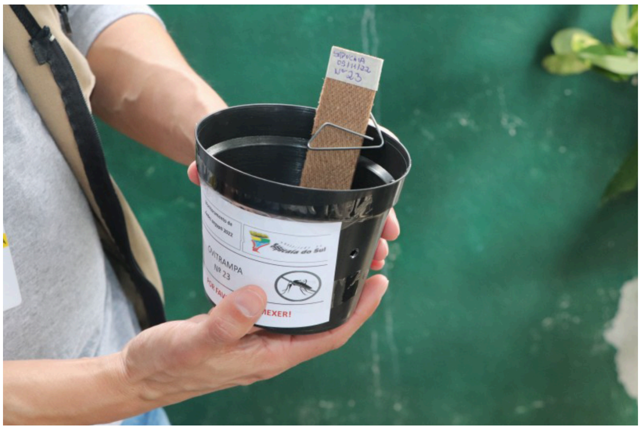
Graph 1. Comparison of confirmed dengue cases: Brazil vs. World



Source: Adapted from World Health Organization, 2014.

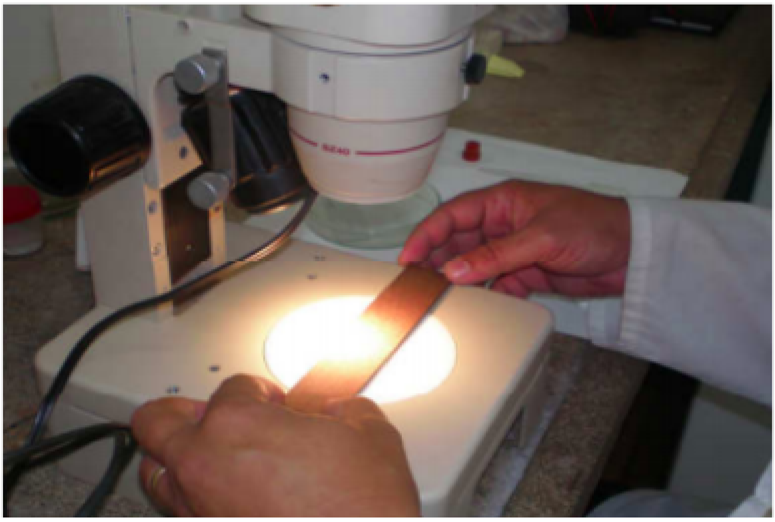
Given the scenario, mosquito surveillance and population control have become critical public health priorities, led by institutions such as FIOCRUZ-PE and regional health surveillance agencies. A common method for monitoring these populations is through the use of ovitraps - devices designed to collect mosquito eggs for counting and analysis. Typically, the egg-counting process is conducted manually by health technicians, sometimes without magnification tools, making it labor-intensive and susceptible to human error. As a result, recent research has turned to automation as a means of improving both the efficiency and accuracy of this task.

Figure 1. A ovitrap consisting of a dark recipient and a wooden plaque



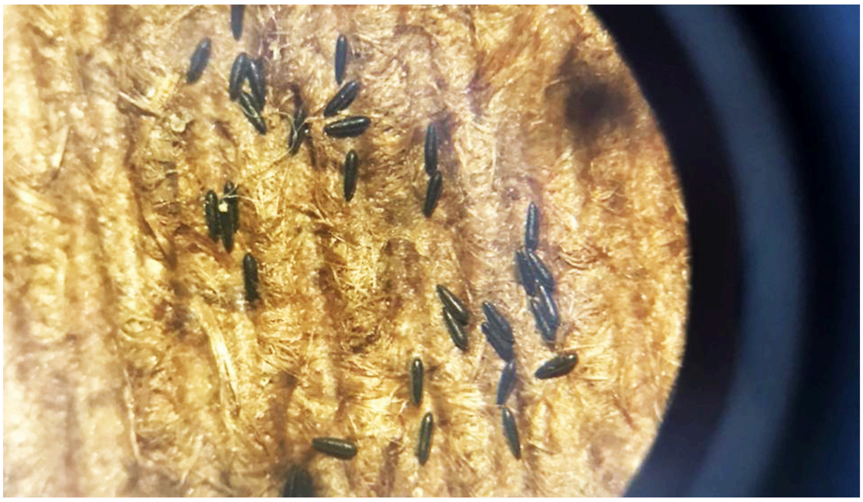
Source: BERNARDES, 2021.

Figure 2. Technician manually counting eggs with a stereomicroscope



Source: BAÊTA, 2007

Figure 3. Mosquito eggs on a wooden plaque, collected from an ovitrap.



Source: BARRETO, 2021.

At the current stage, the available solutions are highly dependent on the quality of the images used, which can be hindered by personal smartphone cameras. Result's accuracy is significantly impacted by image quality, as blurry or unclear images complicate accurate egg counting. This criterion makes it inaccessible to real-life, in-field professionals who could benefit from research. To address this challenge, this research explores the use of image restoration deep learning models to enhance ovitrap images captured by smartphones. By systematically training and evaluating these models using a dataset consisting of close-up, real-world images of mosquito egg pallets, this study seeks to identify different methods for mitigating visual defects and improving overall analysis reliability.

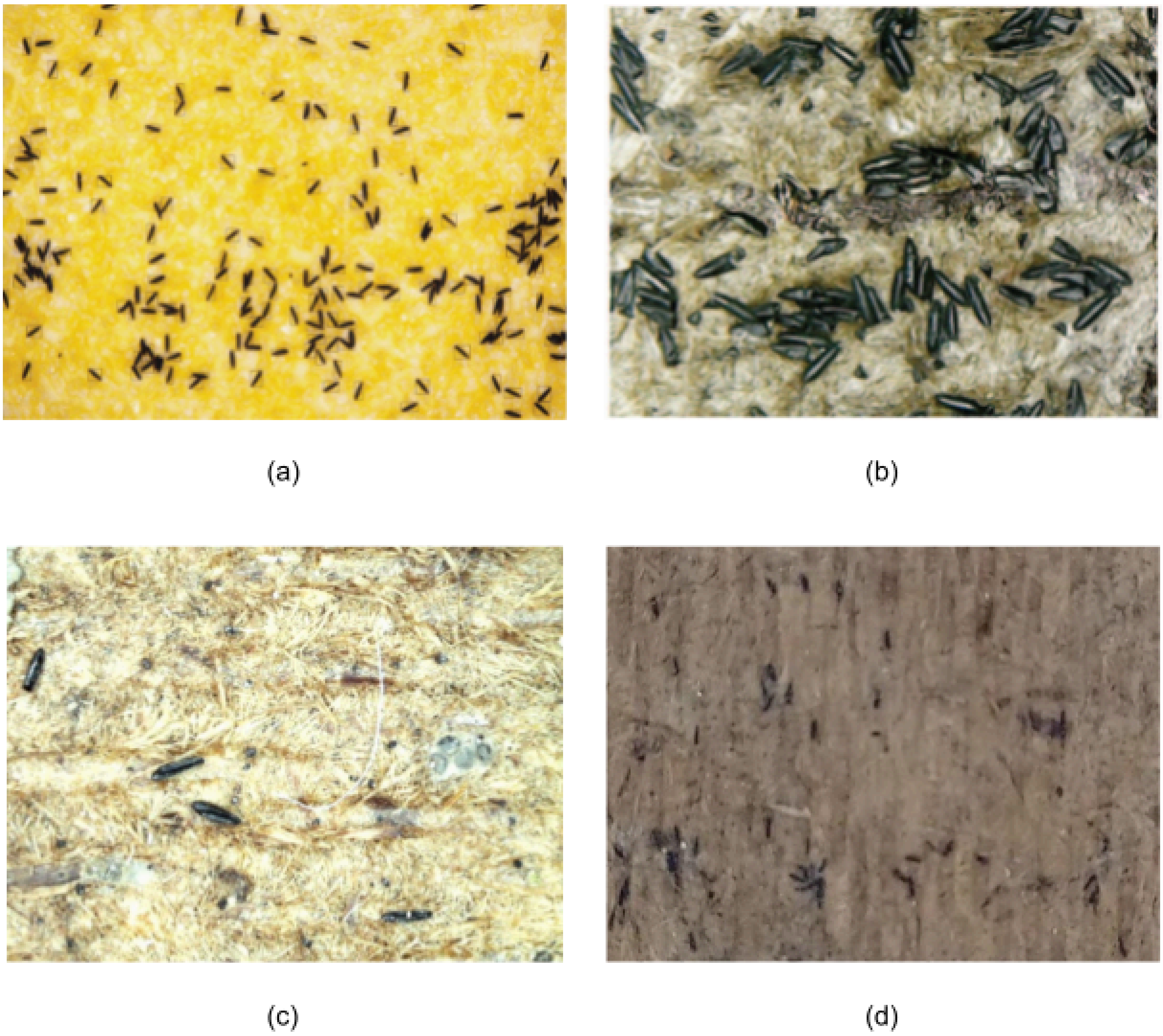
This study is structured as follows: Chapter 2 presents the background and related works in this field by presenting the progress achieved and its limitations. Chapter 3 details the methodology, including dataset preparation, image restoration model architectures and evaluation metrics. Chapter 4 presents the results of experimentation and comparative analyses. Chapter 5 summarizes the results, discussing the implications, limitations, and avenues for future research.

2. RELATED WORK

Numerous research efforts have been dedicated to facilitating the process of counting Aedes *aegypti* eggs with image detection over the recent years. This section highlights the main contributions in this field and relates them to this study's goals and methodology.

Several studies have successfully applied convolutional neural networks (CNNs) to automate mosquito egg counting. Most notably, Javed et al. (2023) proposed EggCountAI, a robust counting tool based on Mask Region-based CNN that achieved up to 98% accuracy when counting Aedes eggs. Similarly, Santana (2019) developed a system using deep learning to recognize Aedes *aegypti* and A. *albopictus* eggs in field conditions, demonstrating the feasibility of deploying AI models for health surveillance.

Rodrigo da Silva's 2021 research was a foundational reference for this work. Silva designed a low-cost system for automated egg counting based on deep learning, incorporating YOLOv10 for object detection. While Silva achieved high accuracy (92%), his solution, like the others previously mentioned, required a carefully controlled setup and depended on consistent image quality.



(a)

(b)

(c)

(d)

Figure 4. Comparison of images from related works: (a) image from Javed et al. (2023); (b) image from Santana (2019); (c) image from Silva (2021); and (d) a lower-resolution image captured by smartphone for the present research.

Recent efforts have focused on building more flexible systems that operate effectively with lower-quality images, such as those captured by smartphones in the field. Wang et al. (2024) introduced a two-step approach using the Segment Anything Model (SAM) and Faster Region-based CNN for identifying and counting Aedes *albopictus* eggs. Their work reached a precision of 0.997 in egg count correlation compared to manual annotations, confirming the feasibility of high-accuracy automated counting in uncontrolled environments.

Outside the mosquito-specific domain, image quality improvement has shown substantial impact on counting accuracy across various fields. In particular, Huynh et al. (2024) integrated super-resolution into a YOLOv4-based system for small object detection, demonstrating that enhancing input image quality directly contributed to improved object detection accuracy. This approach mirrors the strategy used in this study: improving smartphone-captured ovitrap images before object detection.

These recent studies validate the hypothesis that pre-processing images through enhancement techniques can substantially improve the performance of object detection models in egg counting. The current work extends these findings by applying restoration models (Real-ESRGAN, MPRNet, and Restormer) directly to smartphone ovitrap images.

2.1 IMAGE RESTORATION MODELS

The decision to use image restoration deep learning models in this research comes from the necessity to address common issues observed in real-life images captured by smartphones, such as motion blur, inadequate focus, and other limitations inherent in mobile camera technology. Image restoration models are designed specifically to mitigate these issues

Initially, several prominent image restoration models were considered for evaluation. The criterion for consideration included disponibility of source code, ease to execute training and inference, computational resources and duration required and results described in source paper. Ultimately, three models were ultimately selected for experimentation in this research.

Table 2. Comparison of benchmarks on image restoration tasks.

Model	PSNR (dB)	SSIM	Dataset	Task
MPRNet	32.66	0.95	GoPro	Deblurring
RealESRGAN	24.32	0.735	RealSR (real-world)	Real-ESRGAN
Restormer	25.98	0.81	DPDD	Defocus Deblurring
Restormer	32.92	0.96	GoPro	Motion Deblurring

Source: Adapted from ZAMIR et al, 2021; ZAMIR et al, 2022; LI, 2025.

2.1.1 Multi-Stage Progressive Image Restoration (MPRNet)

Published in 2021 by Zamir et al., the Multi-Stage Progressive Image Restoration (MPRNet) paper proposes a multi-stage restoration framework designed to tackle image degradation problems such as deblurring, deraining, and denoising. This architecture progressively learns restoration functions through multiple stages, effectively breaking down the complex restoration process into manageable sub-tasks. The first stages of MPRNet utilize an encoder-decoder structure to capture extensive contextual information, while the final stage operates directly at the original image resolution to preserve spatial details.

A notable advantage of MPRNet, particularly in its image deblurring method, is its ability to balance high-level contextual understanding and detailed spatial accuracy. The framework employs supervised attention modules between stages, which refine intermediate features using ground-truth guidance, thereby ensuring effective feature propagation and restoration accuracy. Additionally, cross-stage feature fusion mechanisms enhance information flow, preserving critical contextual details across stages.

However, despite its strong performance, MPRNet presents some limitations. The complexity increases with each additional stage, affecting runtime, especially in resource-constrained environments, making it potentially less efficient for real-time applications and use on resource-constrained devices. Additionally, the supervised attention module's effectiveness strongly relies on the availability of accurate ground-truth images, which might not always be accessible in real-world scenarios.

It is important to note that while MPRNet was trained in this study to adapt to our custom dataset, the model does not necessarily require training to be used. Pre-trained versions are available and can be used directly for inference. The original MPRNet model was trained on several benchmark datasets depending on the restoration task.

MPRNet is a supervised learning model, meaning it is trained using paired data (i.e., degraded images and their corresponding high-quality reference images). During training, it learns to minimize reconstruction errors by comparing its outputs to these references.

For motion deblurring, the GoPro dataset was used. This dataset contains real-world high-resolution videos captured using a GoPro Hero5 Black camera. Frames from these videos are averaged to create blurred images, paired with sharp ground-truth frames. This dataset is widely used for benchmarking single-image motion deblurring due to its realistic motion blur and diverse scenes. MPRNet is trained in a supervised manner using these paired blurry/sharp images.

2.1.2 Real-ESRGAN

The model was released in 2021 by Wang et al. and it is designed to perform real-world blind image super-resolution (SR), It addresses images with complex, unknown degradations. Its key goal is to improve the realism and clarity of low-quality images, simulating real-world degradation processes more effectively through synthetic training data

Real-ESRGAN utilizes a high-order degradation model instead of the classical first-order model. For training, it employs in sequence the addition of blur, noise, downsampling and JPEG compression to mimic real-life degradation scenarios. The model also integrates sinc filters to emulate common real-world artifacts such as ringing and overshoot. The training process of Real-ESRGAN involves two sequential stages:

1. Real-ESRNet Training: Initially, a Peak Signal-to-Noise Ratio (PSNR) oriented model, named Real-ESRNet, is trained using the L1 loss function. This model serves as a foundation to capture basic image restoration capabilities.
2. Real-ESRGAN Training: In sequence, the weights from the Real-ESRNet training serves as an initialization for the Real-ESRGAN training phase, which employs a combination of L1 loss, perceptual loss (JOHNSON, 2016), and GAN loss (GOODFELLOW, 2014). This sequential step further enhances the model's ability to produce visually appealing and detailed restorations by optimizing perceptual quality.

Nevertheless, the Real-ESRGAN model has certain limitations. It is a heavy network, so the extensive degradation modeling and the complexity of training synthetic datasets can lead to higher computational demands and longer training periods. Additionally, the model may occasionally produce unnatural textures or amplify certain degradation patterns.

Like the other models tested, it does not require training to function; pre-trained weights are publicly available for immediate use. It is a supervised model trained on synthetic datasets generated by applying degradations to high-quality images. The training consists of two stages: first, a PSNR-oriented network is trained using L1 loss; then a GAN-based version is fine-tuned with perceptual and adversarial losses to enhance realism.

Real-ESRGAN was trained using synthetic datasets generated from high-quality images to simulate real-world degradations. Specifically, it was trained on the DF2K dataset, which is a combination of DIV2K and Flickr2K datasets, consisting of high-resolution natural images. These images were synthetically degraded by applying random combinations of blur, noise, downsampling, and compression to mimic realistic image quality issues. Although the degradations are synthetic, the dataset offers a wide variety of real-world textures, lighting conditions, and content types, making it suitable for blind super-resolution training.

2.1.3 Restormer

Released in 2021 by Zamir et. al, the Restormer (Restoration Transformer) framework is designed to perform efficient and effective high-resolution image restoration by addressing limitations of CNN-based methods using Transformers architectures. It incorporates a design for a multi-head attention mechanism and feed-forward neural network to effectively capture long-range pixel interactions while maintaining computational efficiency. It provides solutions such as image deraining, single-image motion deblurring, defocus deblurring (single-image and dual-pixel data), and image denoising (Gaussian grayscale/color denoising, and real image denoising).

In this research, Restormer's capabilities in single-image motion deblurring and defocus deblurring (single-image data) were specifically utilized. The motion deblur method targets images degraded by camera or subject motion, effectively recovering sharp details by modeling dynamic motion patterns. In contrast, the defocus deblur approach addresses images degraded by out-of-focus blurring, restoring fine textures and structural details.

The architecture employs a strategy where it trains on progressively larger patches, effectively capturing global statistics from high-resolution images. It introduces the Multi-Dconv Head Transposed Attention (MDTA) architecture to compute attention across feature channels rather than spatial dimensions; this approach drastically reduces computational complexity. Additionally, it incorporates the Gated-Dconv Feed-forward Network (GDFN), introducing depth-wise convolutions and gating mechanisms that selectively emphasize crucial image features, enhancing detail preservation and image clarity.

Despite its strengths and focus on improving Transformers models, Restormer still is a complex structure. It requires careful optimization especially when used in resource-limited environments. The model's effectiveness depends on a specific training strategy, where patch sizes increase over training epochs; deviations from this strategy may degrade performance. Furthermore, despite demonstrating strong generalization, Restormer can struggle with severely out-of-distribution degradations or unusual blur patterns, occasionally introducing artifacts or overly smoothed textures in challenging cases.

Similar to the other chosen models, Restormer can be used without training by using pre-trained weights made available by the authors. When training is performed, it follows a supervised learning approach, requiring pairs of degraded and high-quality images. The architecture is optimized through a curriculum that gradually increases input patch sizes during training, improving performance on high-resolution data.

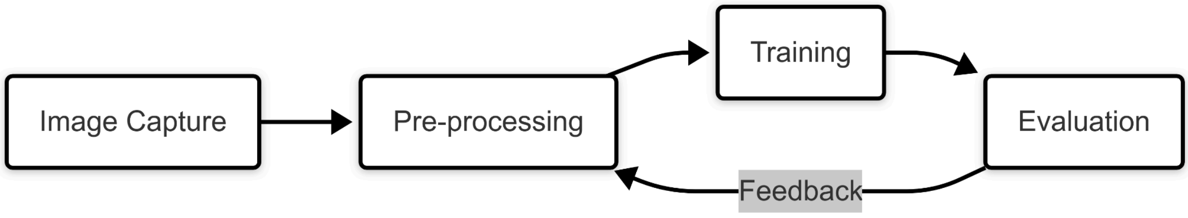
The original Restormer was trained and evaluated on multiple datasets tailored for specific image restoration tasks. For motion deblurring, it used the GoPro dataset, which contains real-world motion blur scenarios. For defocus deblurring, it used the DPDD dataset (Dual-Pixel Defocus Deblurring), which provides paired blurry and sharp images captured using a dual-pixel camera to simulate realistic defocus blur.

3. METHODOLOGY

This study adopted a structured methodology to guide the experimental procedures. An overview of the process is presented in Fluxogram 1, which illustrates the five interconnected phases: image capture, pre-processing, training, evaluation, and feedback.

The process began with image acquisition, followed by pre-processing to refine and structure the dataset according to the requirements of the deep learning models. The data was then used for training and initial evaluation. Based on the outcomes of this evaluation, a feedback loop was established, prompting a second iteration of pre-processing, training, and evaluation to enhance the model's performance.

Fluxogram 1. Workflow of the methodology adopted in the study



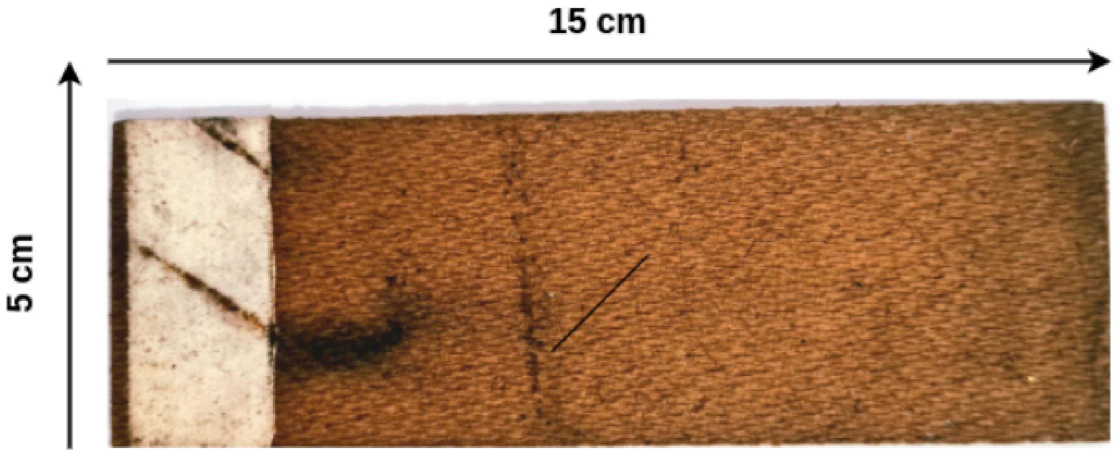
Source: Author, 2025.

Each step will be discussed in detail in the following sections of the current Chapter.

3.1. DATASET COLLECTION

The dataset utilized in this study consists of images of mosquito egg collection plates, also known as ovitraps. Ovitraps consist of dark containers with wide openings, partially filled with water, and a rough wooden paddle installed vertically inside them (GOMES, 2003). A total of 130 ovitrap plates were provided for this research. Each plate is rectangular and designed specifically for egg collection.

Figure 5. Ovitrap plaque and its dimensions



Source: SILVA, 2021

Images were captured using a Samsung Galaxy A03s smartphone, equipped with a primary camera featuring a 13 MP sensor, an f/2.2 aperture lens and autofocus capability. To ensure consistency and stability during image capture, a custom-built support device was provided. The support consists of two distinct components:

1. The first component is similar to a table and securely holds the smartphone horizontally. In its interior, it integrates an internal chamber with built-in lighting to illuminate the image evenly.

2. The second component is similar to a plate and serves as the base and precisely matches the dimensions of the ovitrap plaques. This base contains grooves strategically placed to divide each plate into three equal sections. By adjusting the position of the upper part of the support device, the smartphone camera consistently captures one-third of the plate per image.

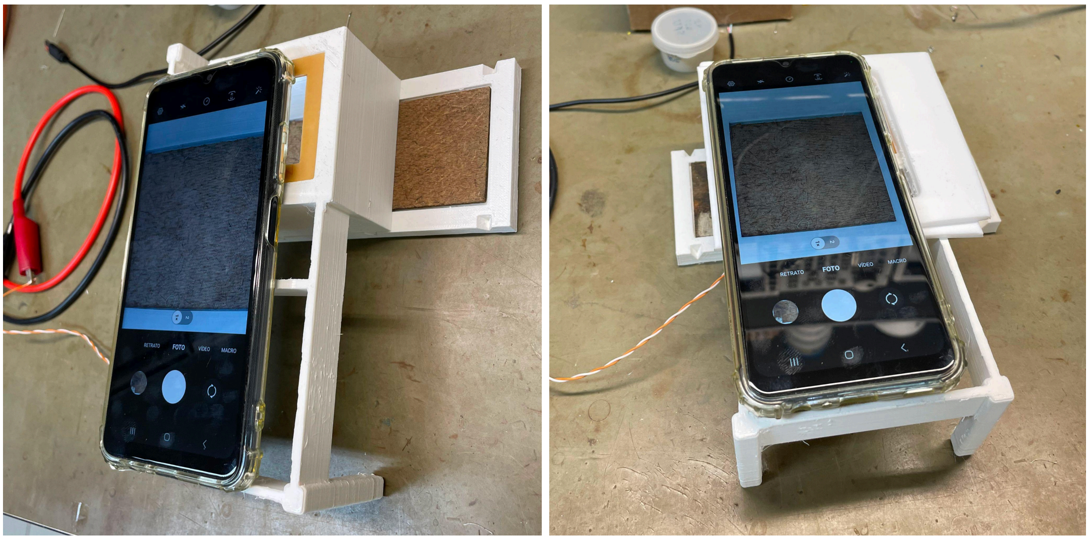


Figure 6. Setup to capture ovitrap plaque images

Using this method, three distinct images were captured for each ovitrap plate, yielding a total of 390 images. After initial data collection, a manual selection process was conducted to identify images suitable for training. Selection criteria included the presence of at least one mosquito egg, adequate focus in any part of the image, and minimal interference from debris such as mud stains, hair, or other artifacts. Based on these criteria, 82 images were deemed suitable and selected for training.

3.2 DATASET SETUP

Table 1. Summary of training and testing image pairs after cropping strategy

Training round	Number of image pairs (Training)	Number of image pairs (Testing)
1	70	12
2	1199	133

To enable the training of supervised learning models, the dataset was organized into paired images — one low-quality input and one high-quality reference. This pairing was necessary because two out of the three selected models (MPRNet and Restormer) rely on supervised learning provided by the user, which requires a degraded image and its corresponding clean version to perform effective training.

Two distinct rounds of training were carried out, each employing a different image cropping strategy. In both training rounds, central image crops were deliberately chosen as the input (degraded) images, while the peripheral crops (corners and edges) served as reference targets. This design choice was made because the center regions of the photos consistently showed reduced focus, likely due to camera lens limitations.

In the first round of training, each original image was divided into nine sections (Figure 4). For training pairs, clear corner crops were selected as reference targets while center crops were selected as inputs. This approach prioritized image quality improvement. The process created 82 pairs of images. As shown in Figure 4, crop A serves as the reference for crop 1.

A more detailed approach was implemented for the second round of training (Figure 5), dividing each image into sixteen equal parts. The four center sections were reserved to act as training inputs. Corresponding corner sections were selected as references. This expanded the dataset to 1332 pairs of images. For demonstration, in Figure 5, crop 1 references crop A, crop 2 references crop B, and so on.

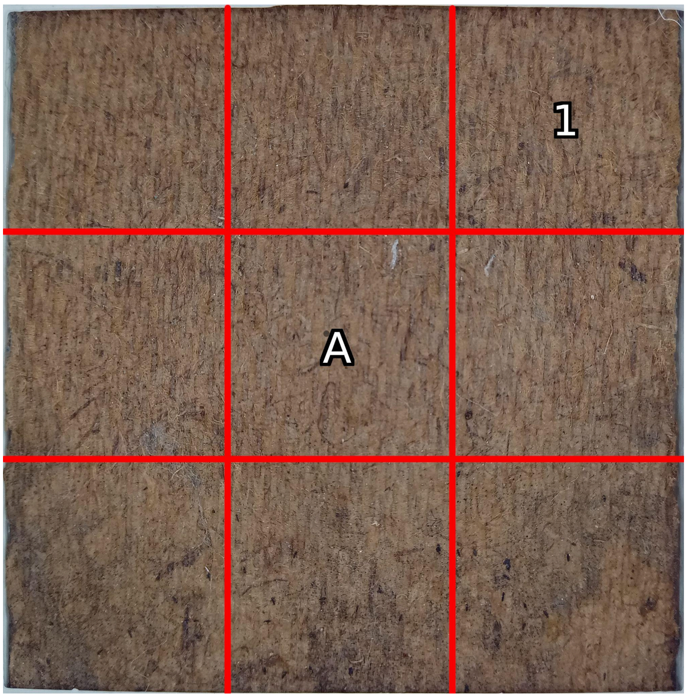


Figure 7. Representation of cropping and selection for the first round of training.

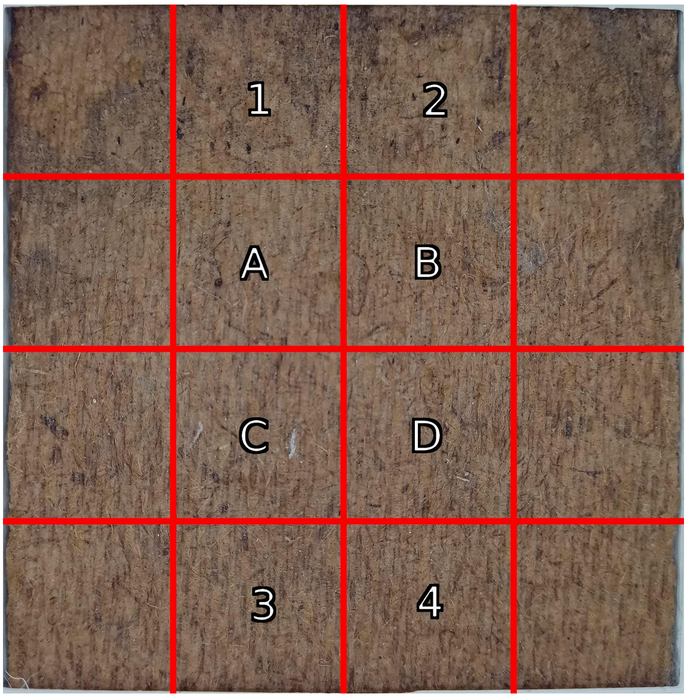


Figure 8. Representation of cropping and selection for the second round of training.

3.3. EXPERIMENTATION RESOURCES

The experiments were conducted on two different, but complementary computational resources. Kaggle[1], a platform that provides free computational resources for data scientists, including a Jupyter Notebook feature and access to GPUs. For this research, the NVIDIA Tesla P100 GPU was primarily used, which allowed efficient computation and accelerated training. However, Kaggle enforces a weekly usage limitation of 40 hours per user, making continuous or extensive training challenging.

To supplement this resource, a shared university clustered virtual machine called Apuana[2] was also employed. The Apuana virtual machine houses NVIDIA A100 80GB PCIe GPUs and NVIDIA GeForce RTX 3090 GPUs and 5,24 TB RAM (ASCOM, 2023). Although it has more GPUs available, institutional restrictions permit the use of no more than two GPUs simultaneously.

[1] https://www.kaggle.com/
[2] https://apuana.cin.ufpe.br/

3.4 EVALUATION METRICS

To evaluate the effectiveness of the image restoration models, one of the metrics chosen was the Peak Signal-to-Noise Ratio (PSNR). PSNR is widely used as a benchmark for quality in image processing tasks. In practice, it is calculated from the mean squared error (MSE) between the restored image and a reference, where *I* denote the image (1).

$$PSNR = 10.log_{10}\left(\frac{max(I)^2}{MSE}\right) \tag{1}$$

A higher PSNR (in dB) indicates the restored image is closer to the ground truth in pixel intensity – thus, a higher PSNR is better. PSNR is useful for quantifying overall fidelity and is easy to compute, but it correlates imperfectly with perceived visual quality, since it treats all pixel errors equally and neglects human visual system characteristics.

While PSNR requires a reference image for its calculation, NIQE does not. The Natural Image Quality Evaluator (NIQE) is a blind image quality assessment metric that evaluates image quality based on a statistical model of natural image patches, measuring deviations from naturalness (MITTAL, 2013). Lower NIQE scores indicate higher perceptual image quality, meaning the image more closely resembles natural image statistics.

In addition to the mathematical comparison, the evaluation also consists of assessing the accuracy of mosquito egg count after image inference. Six ovitrap plates were provided—representing approximately 4.6% of the full dataset of 130 plates—and were counted by trained professionals from the health institution Fiocruz. This data serves as a base for the comparison and evaluation of the effectiveness after training.

3.5 AUTOMATED MOSQUITO EGG COUNTING SYSTEM

The automated mosquito egg counting system used in this study was developed using the source code from Silva (2021), employing the YOLOv10 object detection model with weights specifically trained for mosquito egg detection.

The counting model operates by initially dividing each image given as input into a 7x6 grid, resulting in smaller sub-images. This approach aims to optimize processing performance and reduce the overall demand for computational resources. Each of these sub-images is then processed by the YOLOv10 model, which has been fine-tuned with pre-trained weights specific to the detection of mosquito eggs. The model performs object detection on each sub-image and returns the number of detected eggs, which is then used for analysis.

The system produces two key outputs: a numerical count of detected eggs for each image, and a set of annotated images where each detected egg is highlighted. The annotated outputs allow for visual inspection to verify the presence of false positives or false negatives, thereby supporting further refinement of the model's accuracy and offering transparency in the automated counting process.

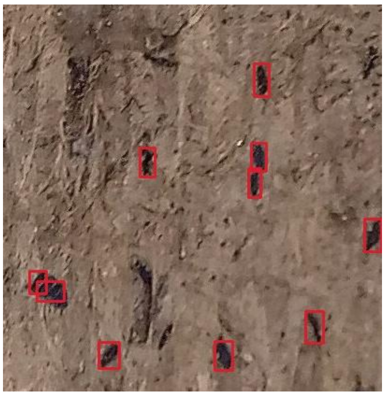


Figure 9. Example of the counting system output. Each red rectangle highlights a mosquito egg detected by the counting system.

Six ovitrap plaques (labeled A to F) were provided by Fiocruz, each manually analyzed and counted by trained professionals. These manual counts served as the ground truth reference for evaluating the performance of the automated mosquito egg counting system under different conditions. All analyses, comparisons, and metrics were based on how closely automated results approached these verified manual counts.

To assess the accuracy of each automated count, a metric called Relative Accuracy was used. This value is calculated by dividing the number of eggs detected automatically by the number counted manually for each individual plaque (2). A relative accuracy of 1.0 means the automated system perfectly matched the manual count; values below 1.0 indicate undercounting, while values above 1.0 suggest overcounting. These scores will be presented throughout the paper in bar charts, with each bar representing the relative accuracy for a specific model and training stage, always in reference to the manual count.

$$Relative\ Accuracy\ =\ \frac{Automated\ Count}{Manual\ Count} \tag{2}$$

This approach allows for a detailed comparison of each image restoration model's performance on a per-plaque basis. By analyzing changes in relative accuracy before and after each training stage, it is possible to identify trends in how different models and training strategies affect detection precision—either improving alignment with the manual baseline or, in some cases, leading to overestimation.

To complement this detailed view, a Total Detection Score was also calculated. This score reflects the overall performance of each model by summing the automated counts across all six plaques and dividing by the total manual count (3). It offers a system-wide perspective on whether the model, as a whole, improves or worsens detection. For instance, unprocessed images resulted in 885 detected eggs out of 1,129 manually counted—about 78.4%. This is the score that will be used for comparison throughout the paper.

$$Total\ Detection\ Score\ =\ \frac{Total\ Automated\ Count}{Total\ Manual\ Count} \tag{3}$$

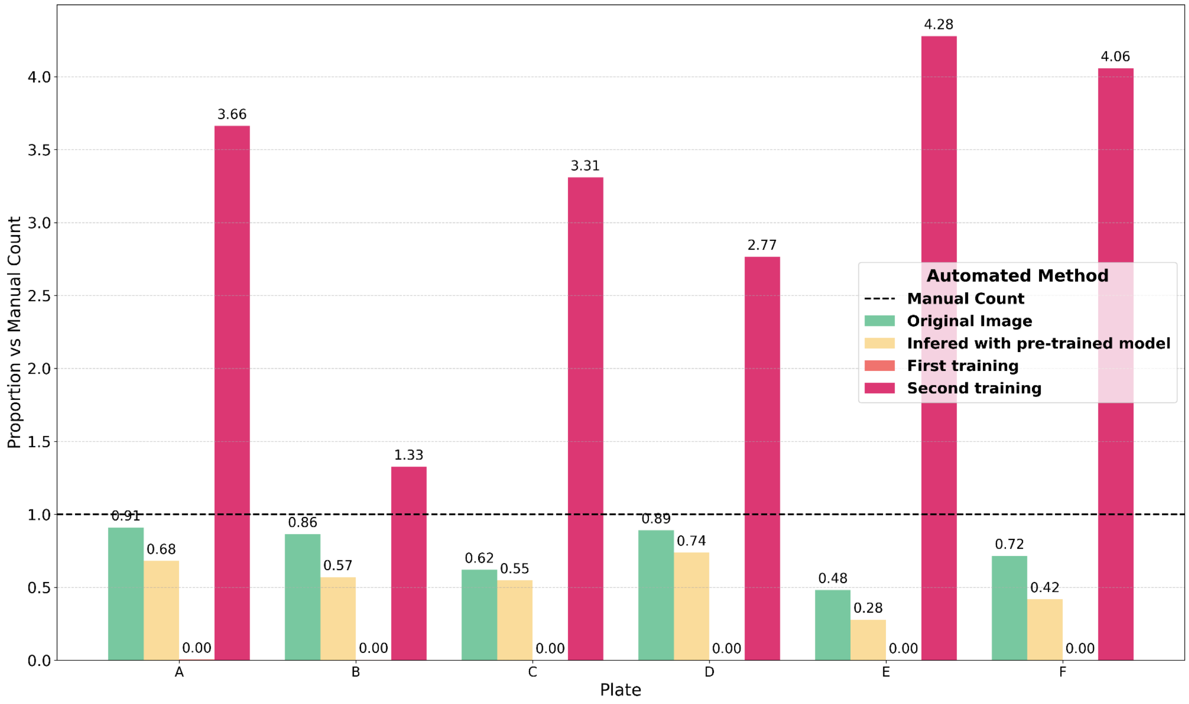
4. RESULTS

Going forward, the results obtained from the experiments with MPRNet, Restormer, and Real-ESRGAN will be discussed. Each model was evaluated based on image quality metrics discussed and the accuracy of automated mosquito egg counting, benchmarked against manual counts performed by trained technicians.

4.1 MPRNET RESULTS

Among the tested models, MPRNet presented the most unstable behavior, showing limited performance improvements in early stages and significant overestimation after training. While initial tests on unprocessed images and pre-trained weights produced counts moderately close to manual results, the custom training iterations caused the model to degrade image quality or generate severe visual artifacts, resulting in either a collapse in detection or an explosion in false positives. In its final form, MPRNet reached a relative accuracy of 331.7%, more than tripling the manual count, which represents the models' generalization to the ovitrap dataset.

Graph 2. Relative accuracy scores by method - MPRNet



Source: Author, 2025

Table 3. Egg counting results using MPRNet at different training stages

Ovitrap plaque	Fiocruz technician manual count	Unprocessed image automated count	Pre-trained model automated count	Automated count after first training	Automated count after second training
A	213	194	145	1	780
B	361	312	206	1	479
C	113	70	62	0	374
D	137	122	101	0	379
E	133	64	37	0	569
F	172	123	72	0	698

Initial testing established baseline performance metrics for MPRNet deblurring training on the available hardware. Using 10 image pairs from the GoPro dataset, each epoch averaged approximately 6 seconds in duration, with individual image inference during about 3 seconds. The training process followed standard protocol with 3001 epochs.

Initially, MPRNet inference was tested using the authors' pre-trained model, originally trained on the GoPro dataset for motion deblurring. Visual inspection of inferred images showed only subtle improvements, mostly in color and brightness, rather than in focus or sharpness. Accordingly, the pre-trained model did not improve counting performance and, in fact, consistently underperformed compared to the original unprocessed images, as shown in Graph 2.

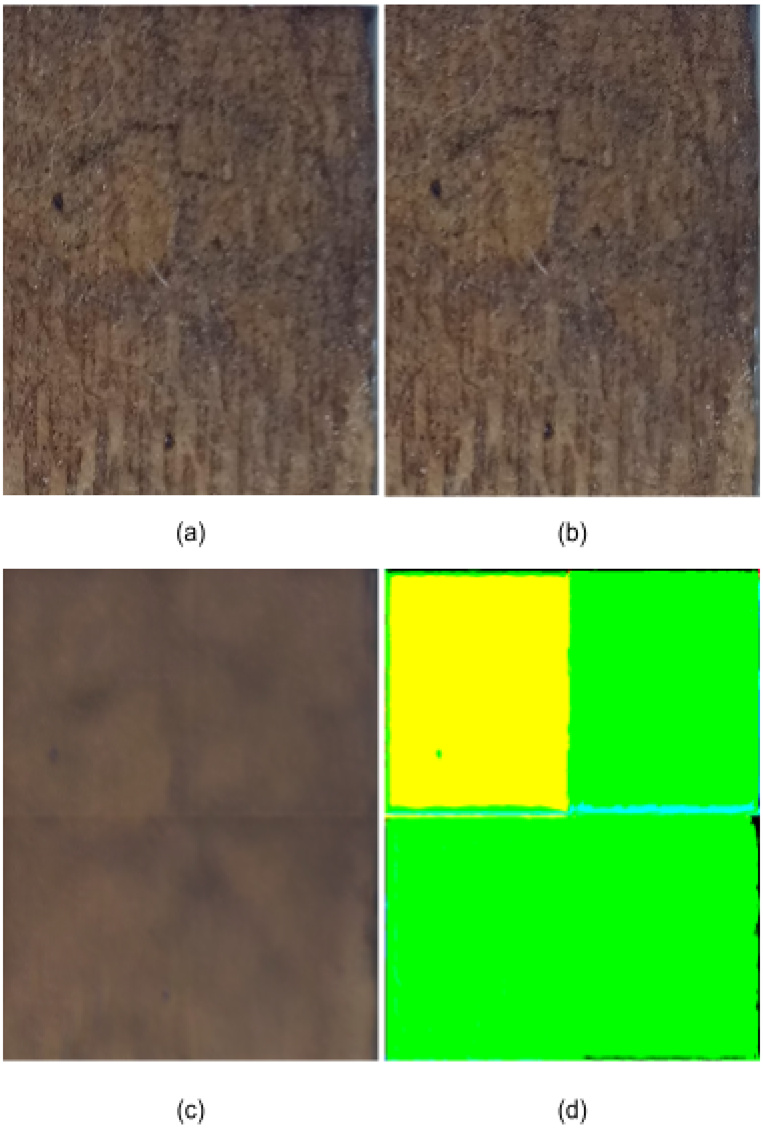


Figure 10. Progression of image enhancement using MPRNet. (a): original image; (b): output using the original pre-trained weights; (c): result after the first training iteration; (d): result after the second training iteration.

To better tailor the model to the characteristics of the ovitrap dataset, a custom training phase was conducted in two rounds. The first training round, using a small dataset of 70 image pairs, lasted 14 hours, with an average epoch duration of 17.1 seconds. Despite a moderate PSNR validation score of 21.54, the resulting images suffered from increased blur and smoothing effects. These distortions led to severe degradation in automated egg detection, with plate-level counts falling dramatically—even to zero in some cases.

The second training round, using an expanded dataset of 1332 training pairs, extended over 6 days and 21 hours, with an average epoch time of 315 seconds. However, the training did not converge successfully: the validation PSNR dropped to 6.70, and the NIQE score worsened significantly to 26.39. Images inferred at this stage exhibited extreme color distortions and unnatural visual artifacts, including blocks of green, yellow, and blue, severely compromising the ability of the detection system to distinguish eggs from the background. Consequently, egg counts spiked unrealistically, leading to inflated totals and a dramatic loss in reliability.

Table 4. Evaluation scores for images processed with MPRNet at different training stages

	No training	First training	Second training
PSNR ↑	n/a	21.54	6.70
NIQE ↓	4.80	9.15	26.39

Graph 2 illustrates the relative accuracy scores by method. While both the original and pre-trained inference results remained relatively close to the ground truth, the first training worsened performance in most cases. The second training produced visually distorted outputs, which the YOLOv10-based detection system interpreted as hundreds of false positives, drastically inflating the egg count.

4.2 REAL-ESRGAN RESULTS

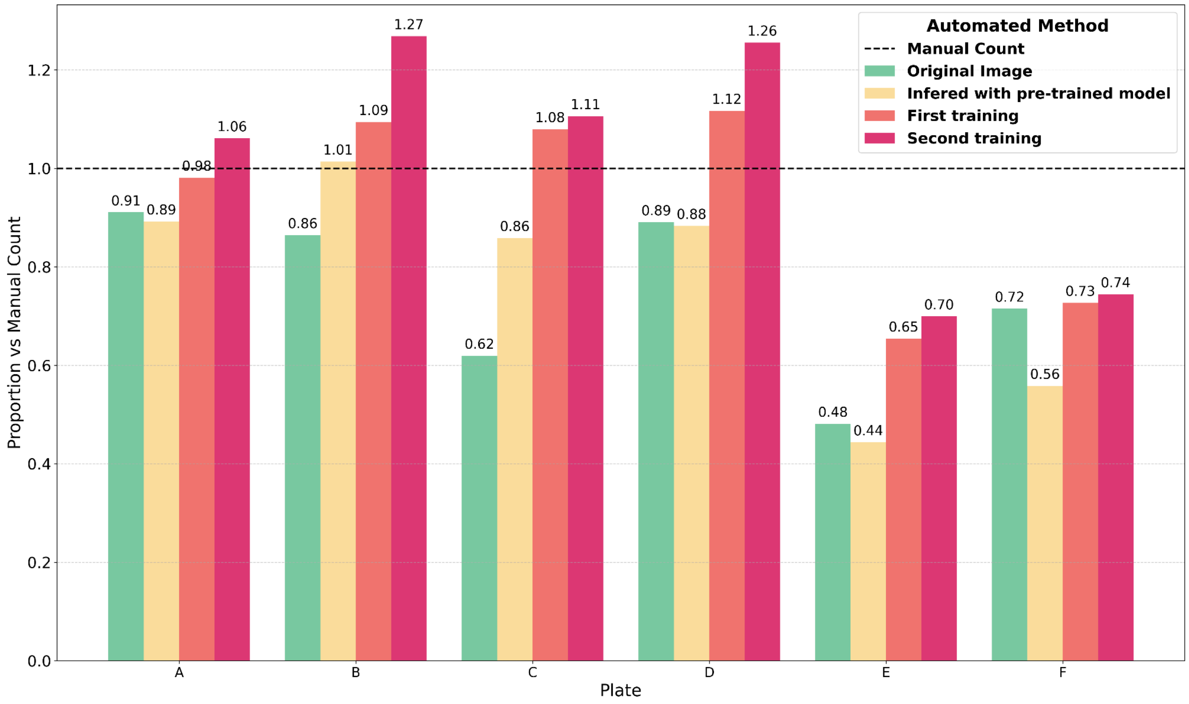
The application of Real-ESRGAN progressively improved the performance of the automated mosquito egg counting system across all tested ovitrap plates. Starting from underwhelming results on unprocessed images, each stage of model refinement—first with pre-trained inference and then with two rounds of custom training—produced increasingly accurate outcomes. The most significant gains were achieved after the second training iteration, where restored images led to counts that not only matched but in some cases exceeded the manual annotations, indicating a strong potential for real-world deployment.

To quantitatively illustrate this improvement, a Total Detection Score was calculated. While the baseline score is 78.4%, the second round of evaluation summed 1,202 eggs, making it a score of 106.5% of the manual benchmark. This increase illustrates both the potential of image restoration to enhance detection and the need to balance sensitivity to avoid overcounting.

Table 5. Egg counting results using RealESRGAN at different training stages

Ovitrap plaque	Fiocruz technician manual count	Unprocessed image automated count	Pre-trained model automated count	Automated count after first training	Automated count after second training
A	213	194	190	209	226
B	361	312	366	395	458
C	113	70	97	122	125
D	137	122	121	153	172
E	133	64	59	87	93
F	172	123	96	125	128

Graph 3. Relative accuracy scores by method - RealESRGAN



Source: Author, 2025

Real-ESRGAN performance was initially evaluated by performing a short training round with a set of 20 images from the DF2K dataset to establish baseline metrics. The total number of iterations selected were the recommended 1,400,000 epochs and results averaged approximately 4 seconds per epoch.

Initially, inference using the authors' pre-trained model, trained on the DF2K dataset, offered only subtle gains in image clarity and marginal improvement in egg detection performance. To better adapt the model to the characteristics of the ovitrap images, a custom training process was initiated.

The first training round, which followed the RealESRNet and RealESRGAN sequential architecture, lasted 11 days and 14 hours, with approximately 1,400,000 iterations and average epochs of 12 seconds. Although this phase enhanced perceptual sharpness and edge definition around the mosquito eggs, the NIQE score slightly worsened (from 4.1 to 7.26), and visual noise was introduced. Nonetheless, it led to improved detection compared to the pre-trained model.

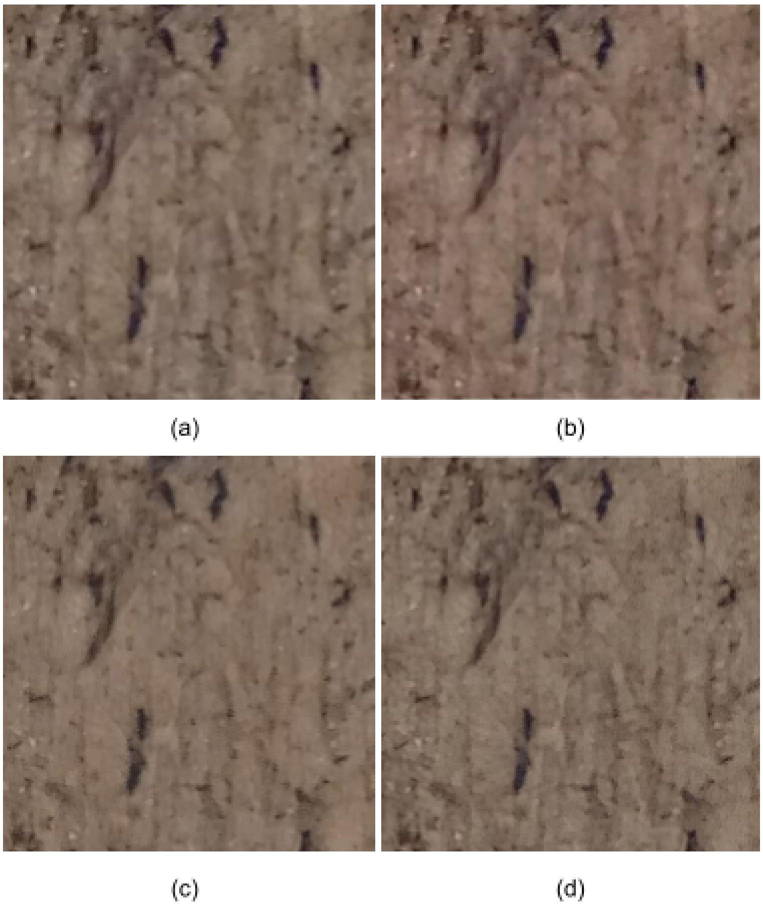


Figure 11. Progression of image enhancement using RealESRGAN. (a): original image; (b): output using the original pre-trained weights; (c): result after the first training iteration; (d): result after the second training iteration.

A second, optimized training round was conducted, lasting 8 days and 13 hours. Even though epochs averaged 221.15 seconds, the inherent early stop mechanism was activated resulting in a total number of epochs around 2800. This round yielded the most promising results, with the NIQE improving to 4.42 and visible gains in both image quality and detection alignment with manual counts. The restored images revealed clearer egg boundaries and more defined textures, aiding the automated model in more accurate object detection.

Table 6. Evaluation scores for Real-ESRGAN at each training stage (PSNR not applicable due to blind architecture).

	No training	**First training**	**Second training**
PSNR ↑	n/a	n/a	n/a
NIQE ↓	4.1	7.26	4.42

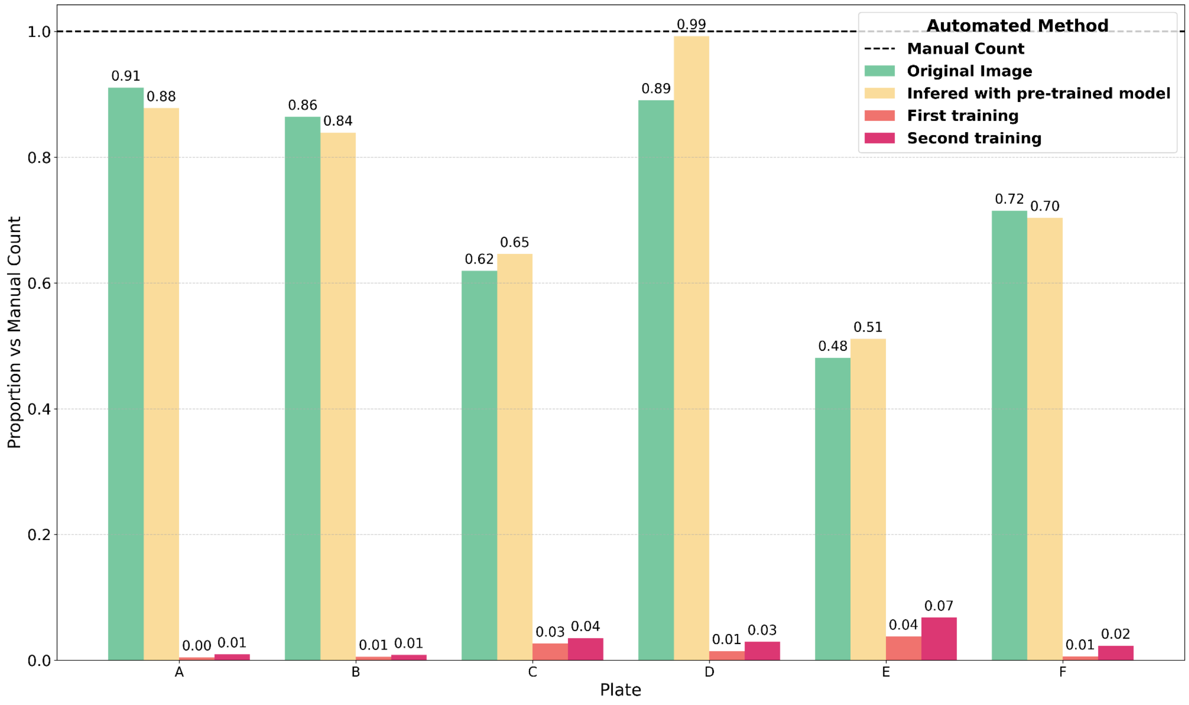
As summarized in Graph 3, relative accuracy scores improved at each stage. While unprocessed images consistently underestimated the number of eggs, results improved with pre-trained inference and continued to rise after each round of custom training. The final model, trained specifically on ovitrap images, consistently outperformed the others and achieved the best balance between visual restoration and detection precision.

4.3 RESTORMER RESULTS

The evaluation of Restormer was conducted in two distinct image restoration tasks: Motion Deblurring and Defocus Deblurring, each trained and tested independently. While the goal was to assess whether either approach could enhance ovitrap images and improve automated egg detection, Restormer demonstrated the lowest overall detection performance among all tested models. While the pre-trained models showed moderate improvements over the baseline in some plates, both the

Motion Deblurring and Defocus Deblurring training paths failed to enhance image quality in a way that supported reliable automated egg detection.

Graph 4. Relative accuracy scores by method - Restormer (Motion Deblur)



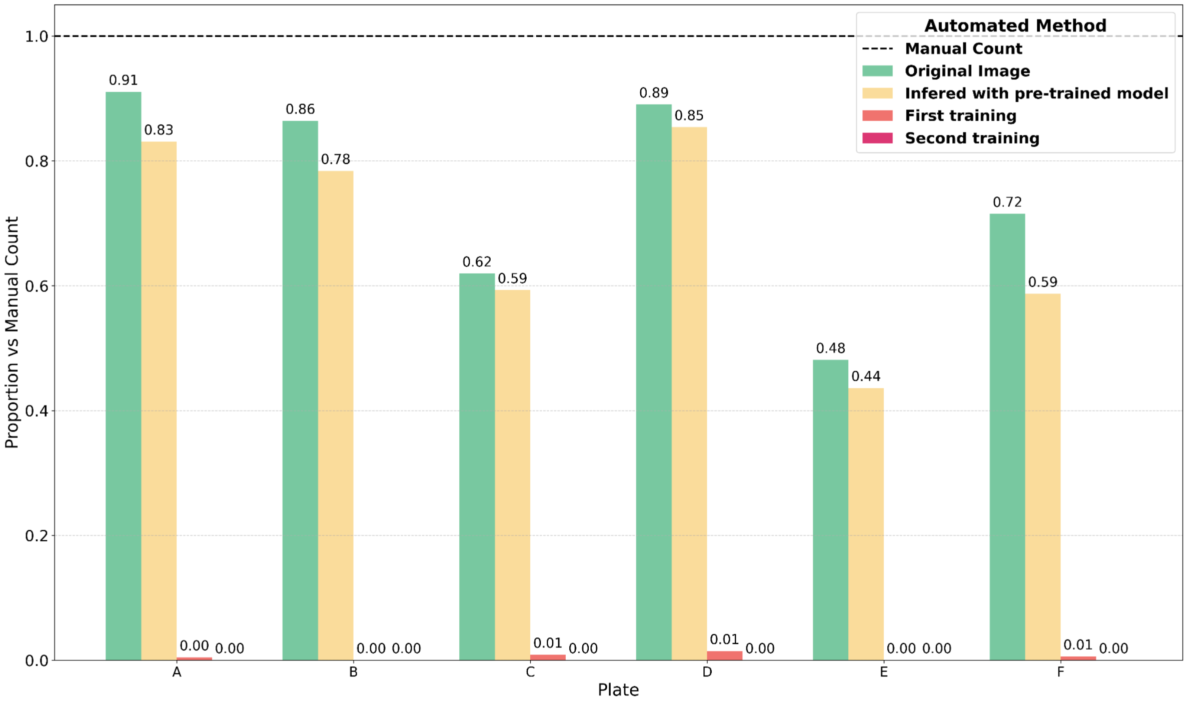
Source: Author, 2025

Table 9. Egg counting results using Restormer Motion Deblur task at different training stages

Ovitrap plaque	Fiocruz technician manual count	Unprocessed image automated count	Pre-trained model automated count	Automated count after first training	Automated count after second training
A	213	194	187	1	2
B	361	312	303	2	3
C	113	70	73	3	4
D	137	122	136	2	4
E	133	64	68	5	9
F	172	123	121	1	4

In the initial evaluation phase, the pre-trained models offered promising results. As shown in Graph 4 and 5, most plates reached relative accuracy scores above 0.80, with some approaching the manual benchmark. In particular, the Motion Deblur pre-trained model achieved up to 99% accuracy on one of the plates. This suggests that, when not retrained, the architecture was able to offer a level of restoration compatible with detection requirements.

Graph 5. Relative accuracy scores by method - Restormer (Defocus Deblur)



Source: Author, 2025

Table 10. Egg counting results using Restormer Defocus Deblur task at different training stages

Ovitrap plaque	Fiocruz technician manual count	Unprocessed image automated count	Pre-trained model automated count	Automated count after first training	Automated count after second training
A	213	194	177	1	0
B	361	312	283	0	0
C	113	70	67	1	0
D	137	122	117	2	0
E	133	64	58	0	0
F	172	123	101	1	0

Initially, both motion and defocus deblur tasks were assessed using the official pre-trained models, trained respectively on the GoPro dataset (for motion deblurring) and the DPDD dataset (for defocus deblurring). To replicate this setup and establish a performance baseline, brief training sessions were conducted using 20 image pairs from each dataset. For the Motion Deblurring task, each epoch required approximately 12 seconds, while for the Defocus Deblurring task, epochs averaged around 7 seconds.

Initial evaluations using the pre-trained weights provided by the authors generated NIQE scores of 4.35 for Motion Deblurring and 3.79 for Defocus Deblurring, serving as reference points for subsequent training improvements.

The first round of training for Motion Deblur lasted 1 day and 22 hours, achieving a PSNR score of 21.64 and NIQE score of 5.89. The second round of motion deblurring training extended to 3 days and 30 minutes, with improved PSNR (22.5) but a decreased perceptual quality (NIQE of 7.26).

Table 7. Evaluation scores for images processed with Restormer Motion Deblur task at different training stages

	No training	First training	Second training
PSNR ↑	n/a	21.64	22.5
NIQE ↓	4.35	5.89	7.26

Table 8. Evaluation scores for images processed with Restormer Defocus Deblur task at different training stages

	No training	First training	Second training
PSNR ↑	n/a	20.99	22.5
NIQE ↓	3.79	5.88	7.24

Defocus deblur first training round required 3 days and 19 hours of training, reaching a PSNR of 20.99 and an NIQE of 5.88. The second training round lasted 1 day and 22 hours, resulting in improved PSNR (22.5) but an increased NIQE of 7.24.

Visual analysis indicated a noticeable degradation in image quality over successive training iterations for both motion and defocus deblur methods. Images became excessively smooth, accompanied by irregular, repetitive grainy patches. Notably, small white tags containing handwritten labels, deliberately excluded from the training data, became significantly distorted post-training while trying to blend unsuccessfully with the background. Overall, both tasks suffered with the dataset provided and had similar degradation patterns.

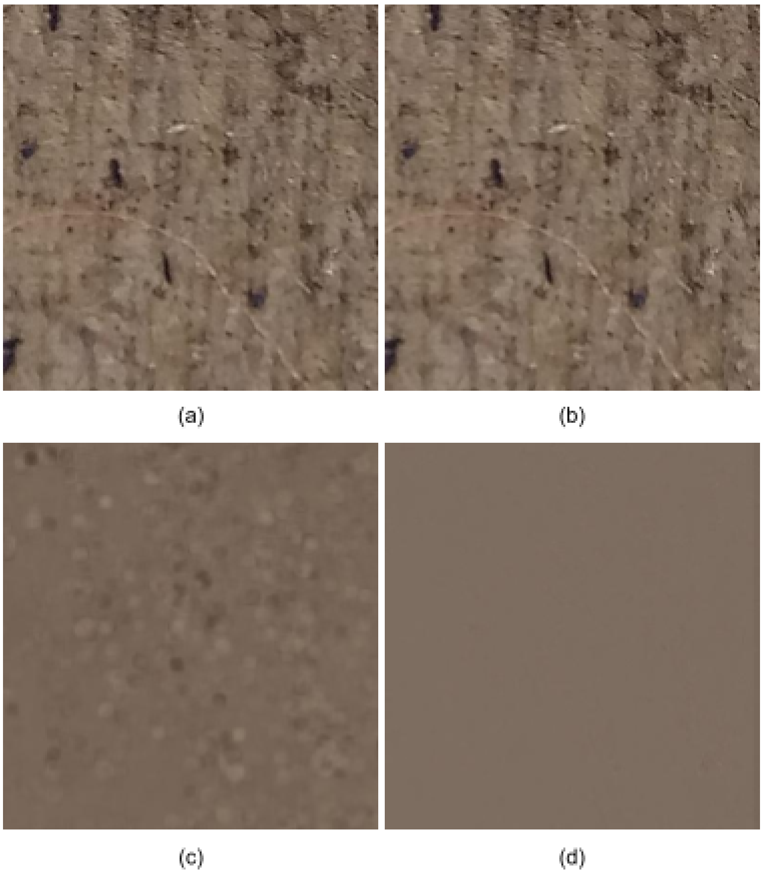


Figure 13. Progression of image enhancement using Restormer Motion Deblur. (a): original image; (b): output using the original pre-trained weights; (c): result after the first training iteration; (d): result after the second training iteration.

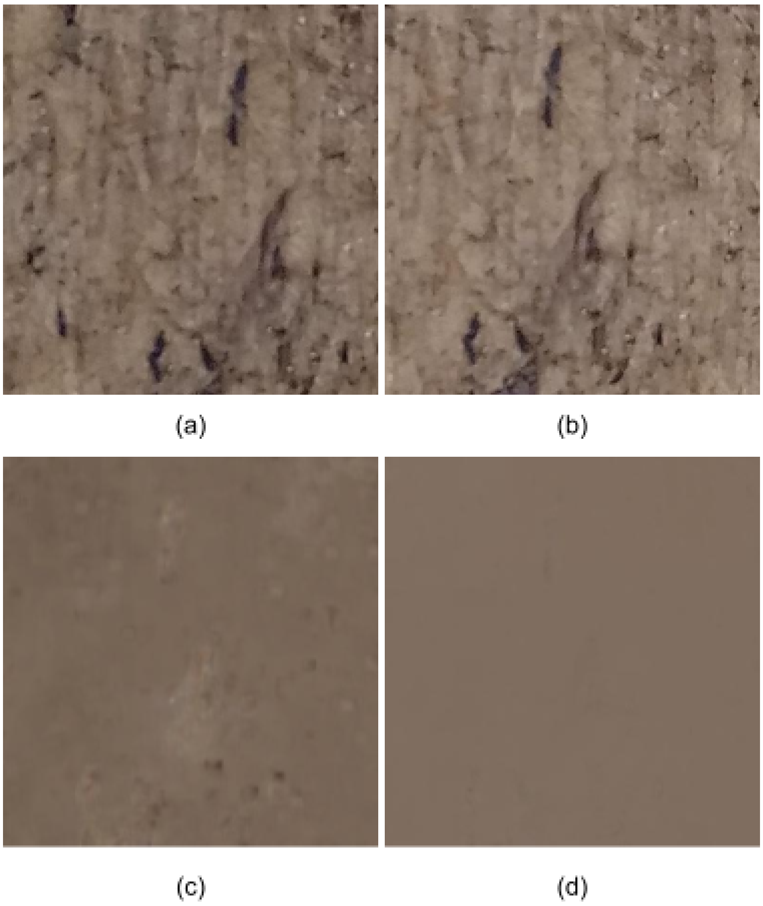


Figure 14. Progression of image enhancement using Restormer Defocus Deblur. (a): original image; (b): output using the original pre-trained weights; (c): result after the first training iteration; (d): result after the second training iteration.

Overall, after processing the images with the two tasks following the second training round, the automated detection system identified only 22 eggs using the Motion Deblur model and 17 eggs using the Defocus Deblur model. These correspond to relative accuracies of just 1.9% and 1.5%, respectively, when compared to the total manual count of 1,129 eggs. These results indicate a critical failure in both training approaches, where the model completely suppressed identifiable structures in its inference, rendering the automated system ineffective.

4.4 RESULTS DISCUSSION

In summary, among the models tested, Real-ESRGAN was the only model to consistently enhance both visual image quality and the accuracy of automated mosquito egg counting, achieving a relative improvement of 28.1 percentage points. Its blind super-resolution approach proved more adaptable to real-world ovitrap images, effectively handling motion blur, defocus, and various noise types without relying on explicit reference pairs. This allowed it to generalize better than models based on supervised training.

In contrast, MPRNet and Restormer did not perform well when retrained with the ovitrap dataset. The pre-trained versions of both models showed moderate performance in some test cases, but once retraining was applied, the models failed to adapt their weights effectively to the characteristics of the new data. MPRNet reached a relative accuracy of 331.7%, significantly overestimating the number of eggs due to the generation of artifacts and distortions in the restored images. Restormer, on the other hand, reached only 1.9% accuracy in its Motion Deblur version and 1.5% in the Defocus Deblur version, underestimating the count and often failing to detect any eggs at all.

These results highlight that models relying on supervised training and reference-based restoration, such as MPRNet and Restormer, may struggle to generalize when applied to datasets with different noise patterns, resolution levels, and structural features than those they were originally trained on. Real-ESRGAN, despite having the longest overall training time among the three models, demonstrated a significant reduction in duration between the first and second training rounds. The adaptability to the dataset, combined with its blind super-resolution strategy, gave Real-ESRGAN greater flexibility, making it more suitable for practical applications in field scenarios with resource constraints and varying image conditions.

5. CONCLUSIONS

This study set out to address a real-world challenge in mosquito surveillance: the manual counting of Aedes *aegypti* eggs, which is labor-intensive, error-prone, and often performed under limited conditions without magnification tools. Recognizing that poor image quality captured via smartphones in the field poses a major barrier to automation, this study proposed and evaluated the application of deep learning-based image restoration models—MPRNet, Restormer, and Real-ESRGAN—to improve the accuracy of automated egg counting systems, focusing on the application of a YOLOv10-based system.

To ground the study in practical applicability, a low-cost and consistent image acquisition method was developed, using a smartphone mounted on a custom support structure to photograph ovitrap plates. The dataset derived from this setup was processed using different cropping strategies to prepare training pairs for deep learning models.

Across the experiments, Real-ESRGAN demonstrated the best overall performance, successfully enhancing both perceptual image quality and object detection accuracy. After two rounds of training using ovitrap-specific data, Real-ESRGAN achieved a Total Detection Score of 1.065, detecting 1,202 eggs compared to the 1,129 manually counted by Fiocruz technicians. This corresponds to a relative accuracy of 106.5%, representing an improvement of 28.1 percentage points over the baseline performance of 78.4%. The model also achieved a NIQE score of 4.42, indicating substantial perceptual quality gains.

In contrast, MPRNet, while theoretically capable of high-quality restoration, failed to generalize to the characteristics of ovitrap images. After training, it produced images with severe color distortions and artifacts that led to an excessive number of false positives, with a final relative accuracy of 331.7%, indicating a collapse in detection precision.

Restormer, evaluated through both Motion Deblurring and Defocus Deblurring tasks, showed promising results in its pre-trained form. However, after training on ovitrap data, both versions experienced a dramatic drop in performance, with final relative accuracies of 1.9% and 1.5%, respectively. This was caused by excessive smoothing and structural degradation in the restored images, preventing the detection system from identifying the mosquito eggs.

This study demonstrates that while certain deep learning models struggled with the challenging dataset (consisting of close-up images containing debris and inexact image pairs), blind super-resolution models like Real-ESRGAN showed effective adaptation when retrained with domain-specific data. The approach offers public health institutions a low-cost, replicable solution for deployment, particularly beneficial for those currently performing manual egg counting under microscopes. By combining smartphone cameras with open-source models, this workflow reduces operational effort, improves consistency, and enhances surveillance efficiency.

Throughout the research, several technical and methodological challenges were encountered. One major limitation was that supervised image restoration models such as MPRNet and Restormer were unable to effectively recalibrate their pre-trained weights when exposed to ovitrap plate images. Even with custom training using paired data, the models failed to converge meaningfully, producing images that either degraded visually or introduced structural noise, ultimately impairing the detection process. Additionally, the recommended training configuration for these models specifies four or more GPUs, which was not feasible during this experimentation. This hardware limitation resulted in significantly extended training durations.

For future research, additional training of Real-ESRGAN on a larger dataset promises to increase the effectiveness of image restoration. Applying this methodology to newer blind image restoration models also shows promise for improved results. A brief experimentation of InstantIR (HUANG et al., 2024) demonstrated visually compelling outcomes without any training, although a complete experiment could not be performed due to time constraints.

Overall, this work highlights the value of adapting machine learning tools to the specific conditions of public health research. By combining deep learning image restoration with automated detection, the study offers a practical and affordable approach to support mosquito egg counting. The results contribute to ongoing efforts to reduce manual workload and improve data reliability in the surveillance of mosquito populations, especially in resource-constrained environments.

REFERENCES

ASCOM. *CIn lança cluster Apuana, infraestrutura de processamento de alto desempenho para execução de pesquisas na área de Inteligência Artificial.* Universidade Federal de Pernambuco, maio de 2023.

BAÊTA, K. F. *Avaliação de armadilhas para monitoramento de culicídeos em aeroporto e portos brasileiros*. 2007

BARRETO, Izinaldo. *Feira participa de pesquisa de monitoramento do Aedes aegypti.* 2021. Available at: https://www.feiradesantana.ba.gov.br/

BERNARDES, Marcelo. *Estado inicia implementação de nova tecnologia no monitoramento de Aedes aegypti*. 2022. Available at: https://saude.rs.gov.br/

GOODFELLOW, Ian; et al. *Generative adversarial nets*. NeurIPS, 2014

GOMES, A.; SILVA, N.; MARQUES, G.; BRITO, M. *Host-feeding patterns of potential human disease vectors in the Paraíba Valley Region, State of São Paulo, Brazil*. Journal of Vector Ecology: Journal of the Society for Vector Ecology, v. 28, n. 1, p. 74–78, 2003

HUYNH, Xuan Nhat; JUNG, Ga Beom; SUHR, Jin Keun. *One-stage small object detection using super-resolved feature map for edge devices*. Electronics, v. 13, n. 2, p. 409, 2024.

HUANG, Jen-Yuan; WANG, Haofan; WANG, Qixun; BAI, Xu; AI, Hao; XING, Peng; HUANG, Jen-Tse. *InstantIR: Blind Image Restoration with Instant Generative Reference*. arXiv preprint, 2024. Available at: https://arxiv.org/abs/2410.06551. Accessed on: 21 mar. 2025.

JAVED, Nouman et al. *EggCountAI: A convolutional neural network-based software for counting of Aedes aegypti mosquito eggs*. Parasites & Vectors, [S.l.], v. 16, n. 341, p. 1–12, 2023.

JOHNSON, Justin et al. *Perceptual losses for real-time style transfer and super-resolution*. European Conference on Computer Vision (ECCV), 2016.

LASERNA, Andrés; BARAHONA-CORREA, Julián; BAQUERO Laura; CASTAÑEDA-CARDONA, Camilo; ROSSELLI, Diego. 2018. *Economic impact of dengue fever in Latin America and the Caribbean: a systematic review.* Revista Panamericana de Salud Pública, v. 42, 2018.

LI, Yachao et al. *StructSR: Refuse Spurious Details in Real-World Image Super-Resolution.* arXiv, 2025. Available at: https://arxiv.org/abs/2501.05777. Accessed on: 20 mar. 2025.

MITTAL, Anish et al. *Making a “Completely Blind” Image Quality Analyzer*. IEEE Signal Processing Letters, v. 20, n. 3, p. 209–212, mar. 2013.

SANTANA, Clodomir José de Souza de Lima et al. *A solution for counting Aedes aegypti and Aedes albopictus eggs in paddles from ovitraps using deep learning*. IEEE Latin America Transactions, v. 17, n. 12, p. 1987–1994, 2019.

SILVA, Rodrigo Emerson Valentim da. *Contagem automática de ovos do mosquito Aedes aegypti utilizando métodos de aprendizagem profunda e dispositivo de baixo custo*. Tese (Mestrado). Universidade Federal de Pernambuco, Recife. 2021

WANG, Minghao et al. *Enhancing vector control: AI-based identification and counting of Aedes albopictus (Diptera: Culicidae) mosquito eggs*. Parasites & Vectors, v. 17, n. 511, p. 1–12, 2024.

WANG, Xintao, XIE, Liangbin, DONG, Chao; SHAN, Ying. *Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data*. IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), out. 2021.

WORLD HEALTH ORGANIZATION. Dengue - Global situation. Available at: <https://www.who.int/emergencies/disease-outbreak-news/item/2024-DON518>. Acessed on 1 mar. 2025

ZAMIR, Syed Waqas; ARORA, Aditya; KHAN, Salman; HAYAT, Munawar; KHAN, Fahad Shahbaz; YANG, Ming-Hsuan; SHAO, Ling. *Multi-stage progressive image restoration*. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), nov. 2021.

ZAMIR, Syed Waqas; ARORA, Aditya; KHAN, Salman; HAYAT, Munawar; KHAN, Fahad Shahbaz; YANG, Ming-Hsuan; SHAO, Ling. *Restormer: Efficient Transformer for High-Resolution Image Restoration*. 1 jun. 2022.