

NHẬN DIỆN SÂU BỆNH HẠI KÍCH THƯỚC NHỎ

Tên đề tài (Tiếng Việt): Nghiên cứu giải pháp phát hiện đối tượng nhỏ (Small Object Detection) ứng dụng trong nhận diện sớm sâu bệnh hại trên cây trồng sử dụng chiến lược Cắt lát ảnh (Slicing Aided Hyper Inference) và Mạng kim tự tháp đặc trưng (FPN).

Tên đề tài (Tiếng Anh - Suggested): *Small Pest Detection in Field Crops: A Coarse-to-Fine Framework using Slicing Aided Hyper Inference (SAHI) and Attention-guided Feature Pyramid Networks.*

1. Đặt vấn đề (Problem Statement)

- Thực trạng:** Sâu hại (bọ trĩ, rầy nâu, nhện đỏ) thường có kích thước rất nhỏ và ẩn nấp dưới tán lá. Việc phun thuốc tràn lan gây ô nhiễm. Nông nghiệp chính xác (Precision Agriculture) cần biết chính xác vị trí và mật độ sâu để phun thuốc cục bộ (Spot Spraying).
- Hạn chế công nghệ:**
 - Vấn đề biến mất đặc trưng (Feature Vanishing):** Các mô hình như YOLO hay ResNet thường giảm kích thước ảnh (Downsampling) xuống 32 lần (Stride=32). Một con sâu kích thước 16×16 pixel sẽ bị nén xuống còn 0.5×0.5 pixel -> Mất hoàn toàn thông tin.
 - Mất cân bằng mẫu (Class Imbalance):** Trong một bức ảnh chụp cái cây, 99% là lá (background), chỉ 1% là sâu (foreground). Các hàm Loss thông thường sẽ bị áp đảo bởi background.
- Câu hỏi nghiên cứu (Research Question):** Làm thế nào để thiết kế một cơ chế suy luận (Inference mechanism) có khả năng bảo tồn thông tin của các đối tượng siêu nhỏ (chiếm $< 0.1\%$ diện tích ảnh) mà không làm tăng chi phí tính toán lên mức không thể chấp nhận được?

2. Mục tiêu nghiên cứu (Research Objectives)

- Tích hợp chiến lược SAHI (Slicing Aided Hyper Inference):** Không đưa toàn bộ ảnh to vào mạng. Thay vào đó, cắt ảnh thành các miếng nhỏ (patches) có độ chồng lấp (overlap), thực hiện nhận diện trên từng miếng, sau đó ghép kết quả lại.
- Tối ưu hóa kiến trúc FPN (Feature Pyramid Network):** Cải tiến các kết nối ngang (lateral connections) để truyền tải thông tin chi tiết từ các lớp nông (shallow layers) xuống các lớp sâu, giúp mạng “nhìn” thấy các chi tiết cạnh/gờ của con sâu nhỏ.
- Xử lý ngụy trang (Camouflage Handling):** Sâu thường có màu giống lá. Cần tích hợp Attention Mechanism (CBAM/SE) để mạng tập trung vào sự khác biệt về kết cấu (texture) thay vì chỉ dựa vào màu sắc.

3. Tổng quan tài liệu & Khoảng trống nghiên cứu (Literature Review)

3.1. Các phương pháp Object Detection tiêu chuẩn:

- **YOLOv5/v8:** Rất mạnh với vật thể trung bình và lớn (người, xe). Tuy nhiên, độ đo mAP (mean Average Precision) giảm thê thảm (thường $< 20\%$) đối với tập dữ liệu “Small Objects” (như MS COCO-Small).

3.2. Các kỹ thuật chuyên trị vật thể nhỏ (Small Object Detection Techniques):

- **Dilated Convolution (Tích chập giãn):** Giúp mở rộng vùng cảm nhận (Receptive Field) mà không cần giảm kích thước ảnh, giữ lại chi tiết không gian [1].
- **Feature Pyramid Networks (FPN):** Đây là kiến trúc xương sống cho bài toán này. Nó kết hợp đặc trưng ngữ nghĩa mạnh (ở tầng sâu) với đặc trưng không gian chi tiết (ở tầng nông).
- **SAHI (2022):** Đây là thư viện “thần thánh” cho ảnh độ phân giải cao. SAHI cắt ảnh thành các lát (slices), detect trên từng lát, rồi dùng thuật toán NMS (Non-Maximum Suppression) để gộp kết quả. Nó giúp tăng mAP cho vật thể nhỏ lên 10-15% mà không cần train lại model [2].

3.3. Khoảng trống nghiên cứu (The Gap):

- **Chi phí tính toán của SAHI:** SAHI rất tốt nhưng rất chậm (vì phải chạy model 10-20 lần cho 1 bức ảnh cắt lát). Chưa có nhiều nghiên cứu về việc *lựa chọn vùng cắt thông minh* (chỉ cắt vùng có khả năng có sâu, bỏ qua vùng trời/đất).
- **Dữ liệu sâu hại thực tế:** Các dataset hiện nay (như IP102) chủ yếu là ảnh cận cảnh (macro). Thiếu các dataset chụp góc rộng (wide-angle) mô phỏng camera giám sát hoặc drone.

4. Tài liệu tham khảo minh chứng (References)

Các bài báo khoa học 2022-2024 cần đọc để viết Proposal:

1. Về kỹ thuật SAHI:

- *Paper:* “Slicing Aided Hyper Inference and Fine-tuning for Small Object Detection” (ICIP 2022). *Đây là bài báo gốc của SAHI, bắt buộc phải trích dẫn.*

2. Về Pest Detection:

- *Paper:* “Small Object Detection for Pest Recognition Using Super-Resolution and Anchor-Free Network” (Computers and Electronics in Agriculture, 2023).
- *Paper:* “Attention-based Feature Pyramid Network for Small Pest Detection in Wild Environment” (IEEE Access, 2024).

3. Về Dataset:

- *Dataset*: “IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition” (CVPR 2019).
- *Dataset*: “Pest24: A Large-scale Dataset for Pest Detection in Field Crops”.

5. Tài nguyên bước đầu (Resources Part 1)

A. Datasets (Dữ liệu):

1. **IP102**: Chứa 75,000 ảnh của 102 loại sâu bệnh. Đây là chuẩn mực (benchmark).
2. **Pest24**: Tập trung vào 24 loại sâu phổ biến, ảnh chụp thực tế ngoài đồng ruộng với nhiều nhiễu.

B. GitHub Repositories (Công cụ):

1. **SAHI (Slicing Aided Hyper Inference)**: github.com/obss/sahi
 - Thư viện này có thể tích hợp với mọi model (YOLOv5, v8, MMDetection). Sinh viên chỉ cần gọi hàm `get_sliced_prediction()`.
2. **YOLOv9 / YOLOv10**:
 - Các phiên bản mới này có kiến trúc “Programmable Gradient Information” (PGI) giúp giữ lại thông tin tốt hơn các bản cũ, rất đáng thử nghiệm cho vật thể nhỏ.

5. Phương pháp nghiên cứu & Kiến trúc đề xuất (Methodology)

Đề tài đề xuất kiến trúc “**Attention-guided Slicing Framework**” (Khung cắt lát định hướng bởi sự chú ý).

Giai đoạn 1: Suy luận toàn cục (Global Inference - The “Glance”)

- **Input**: Ảnh gốc độ phân giải cao (ví dụ: 3000×4000 px).
- **Model**: Sử dụng **YOLOv8-Nano** (phiên bản siêu nhẹ).
- **Resize**: Thu nhỏ ảnh xuống 640×640 px để đưa vào model.
- **Mục tiêu**: Ở bước này, ta chấp nhận mất thông tin sâu bệnh nhỏ. Mục tiêu là phát hiện “**Vùng lá cây**” (ROI) và loại bỏ vùng trời, đất, hoặc khoảng trống.
- **Output**: Một danh sách các tọa độ vùng cần quan tâm (Potential Regions).

Giai đoạn 2: Cắt lát thông minh (Intelligent Slicing - The “Stare”)

Thay vì cắt toàn bộ ảnh theo lưới (Grid) như SAHI mặc định, ta chỉ cắt tại các vùng ROI tìm được ở Giai đoạn 1.

- **Cơ chế cắt**: Tạo các lát cắt (patches) kích thước 640×640 px từ ảnh gốc (High-res) tại vị trí ROI.

- **Overlapping:** Các lát cắt chồng lấp nhau 20% để đảm bảo không có con sâu nào bị cắt đôi ở biên.

Giai đoạn 3: Mạng phát hiện chi tiết (Fine-grained Detection Network)

Các lát cắt được đưa vào mô hình chính.

- **Backbone:** Sử dụng **YOLOv8-S** hoặc **YOLOv9-C** (tùy vào phần cứng).
- **Cải tiến FPN (Feature Pyramid Network):**
 - Thêm các kết nối tắt (Skip Connections) từ tầng P2 (độ phân giải cao nhất trong feature map) để bảo toàn thông tin không gian chi tiết.
- **Module Chú ý (Attention Module - CBAM):**
 - Tích hợp **Convolutional Block Attention Module (CBAM)** vào sau mỗi khối C2f của YOLO.
 - *Tác dụng:* Giúp mạng phân biệt được “con rệp màu xanh” nằm trên “cái lá màu xanh” dựa trên sự khác biệt về kết cấu (vỏ rệp bóng hơn lá).

Giai đoạn 4: Hợp nhất kết quả (Result Merging)

- Sử dụng thuật toán **NMS (Non-Maximum Suppression)** để loại bỏ các bounding box trùng lặp sinh ra do việc cắt chồng lấp.
- Ánh xạ tọa độ từ các lát cắt nhỏ (x', y') về hệ tọa độ của ảnh gốc lớn (X, Y).

6. Kế hoạch thực nghiệm (Implementation Plan)

6.1. Chiến lược dữ liệu (Data Strategy)

Dữ liệu sâu bệnh thường rất mất cân bằng (Imbalanced). Một ảnh có thể có 100 con rầy nhưng chỉ có 1 con sâu ăn lá.

- **Mosaic Augmentation:** Ghép 4 ảnh lại thành 1 để tăng độ phức tạp.
- **Copy-Paste Augmentation (Quan trọng):**
 - Cắt hình con sâu (foreground) từ ảnh này và dán ngẫu nhiên sang ảnh khác (background).
 - *Lợi ích:* Tạo ra hàng ngàn mẫu dữ liệu training mới với các bối cảnh khác nhau mà không tốn công đi chụp. Đây là kỹ thuật SOTA cho bài toán Small Object.

6.2. Metrics đánh giá (Độ đo)

Với bài toán này, chỉ số mAP chung chung là vô nghĩa. Đề tài cần báo cáo chi tiết:

1. **AP_S (Average Precision for Small Objects):** Độ chính xác cho vật thể có diện tích $< 32^2$ pixels. Đây là chỉ số quan trọng nhất.
2. **AP_M (Medium) & AP_L (Large):** Để so sánh.

3. **Inference Time (Tốc độ):** So sánh thời gian chạy của “SAHI gốc” (cắt hết) vs. “Proposed Method” (cắt chọn lọc). Kỳ vọng giảm 50% thời gian.

6.3. Công cụ & Môi trường

- **Training:** Google Colab Pro+ (GPU A100) hoặc máy Server của trường (cần VRAM > 16GB để train với ảnh độ phân giải lớn).
- **Deploy:** Jetson Xavier NX (để demo khả năng chạy trên Drone).

7. Tài nguyên kỹ thuật & Mã nguồn (Resources)

A. GitHub Repositories (Code nền tảng):

1. **YOLOv8 + SAHI Demo:** github.com/obss/sahi/tree/main/demo
 - Đây là code mẫu để chạy SAHI. Sinh viên cần sửa file slicing.py để tích hợp logic “Cắt lát thông minh”.
2. **YOLOv9 (Official):** github.com/WongKinYiu/yolov9
 - YOLOv9 mới ra mắt (2024) có kiến trúc *GELAN* rất mạnh cho việc giữ thông tin. Nên thử nghiệm model này.
3. **CBAM-Attention-YOLO:**
 - Tìm kiếm các repo hướng dẫn cách chèn module Attention vào file yaml cấu hình của YOLO. (Keyword: Add CBAM to YOLOv8).

B. Dataset Links:

- **IP102:** github.com/xpwu95/IP102
- **Pest24:** Dataset này thường nằm trên Baidu Drive hoặc Google Drive do các nhóm nghiên cứu Trung Quốc chia sẻ. Cần tìm kỹ link download.

Lời khuyên:

1. **Đừng Resize ảnh quá nhỏ khi Training:**
 - Sai lầm phổ biến là resize ảnh về 640×640 khi train (theo mặc định của YOLO). Điều này sẽ giết chết các đối tượng nhỏ.
 - *Lời khuyên:* Hãy train với kích thước 1280×1280 (nếu VRAM cho phép) hoặc dùng kỹ thuật **High-Resolution Finetuning**.
2. **Anchor Optimization (Lại là nó):**
 - Sâu bệnh thường có hình dáng dài (sâu) hoặc tròn (bọ). Anchor mặc định của COCO (người, xe) không khớp.
 - Hãy chạy lại thuật toán autoanchor của YOLO trên tập dữ liệu Pest24 để sinh ra bộ anchor mới phù hợp kích thước sâu.
3. **Super-Resolution (Siêu phân giải - Ý tưởng mở rộng):**
 - Nếu camera quá mờ, có thể tích hợp một mạng GAN (như SRGAN/ESRGAN) để tăng độ nét cho các lát cắt trước khi đưa vào detection. Điểm cộng cực lớn cho sự sáng tạo.