



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Nhóm 12

Thành viên nhóm:

Vũ Quang Huy	20173178
Nguyễn Thế Đức	20170057
Nguyễn Minh Hiếu	20173115
Nguyễn Minh Đăng	20172998
Nguyễn Kỳ Tùng	20173455

Nội dung

I. Project 1: Object counting

1. Thuật toán chung
2. Các trường hợp cụ thể

II. Project 2: nhận dạng đối tượng sử dụng mô hình BoW

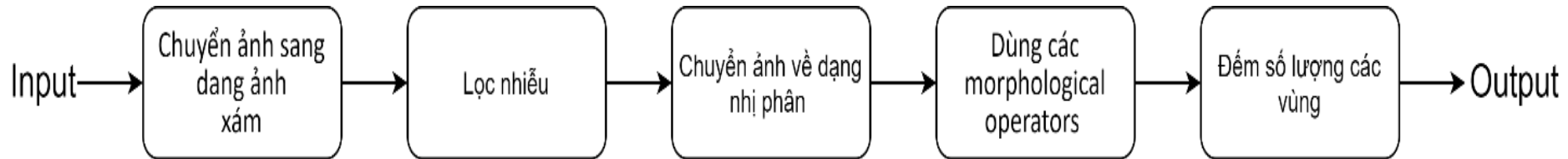
1. Giới thiệu về bộ dữ liệu
2. Mô hình đề xuất
3. Kết quả thực nghiệm
4. Kết luận

III. Tài liệu tham khảo

I. Project 1: Object counting

1. Thuật toán chung

Sơ đồ thuật toán



I. Project 1: Object counting

1. Thuật toán chung

Chi tiết thuật toán:

- Bước 1: Chuyển ảnh sang dạng ảnh xám:
- Bước 2: Sử dụng median filter hoặc lọc trong miền tần số (đối với ảnh có nhiều sinus - Ảnh 3).
- Bước 3: Chuyển ảnh về dạng dạng nhị phân để xác định các vật thể bằng cách sử dụng adaptive threshold:
 - Do ảnh có thể có độ sáng không đồng đều giữa các vùng nên không sử dụng global threshold.
 - Đầu ra là ảnh nền có màu đen, vật màu trắng để xử lý bước tiếp theo, nếu ảnh đầu vào là nền trắng thì tiến hành lấy nghịch đảo pixel.

I. Project 1: Object counting

1. Thuật toán chung

Chi tiết thuật toán:

- Bước 4: Sử dụng các morphological operators (**cv.morphologyEx()**):
 - Loại bỏ các nhiễu bằng opening.
 - Lấp đầy các lỗ trong vật thể bằng closing
- Bước 5: Đếm các vật thể bằng cách đếm các vùng:
Mỗi vật thể sẽ được thể hiện bằng một vùng trắng, chỉ cần đếm các vùng trắng sẽ có được số lượng vật thể bằng hàm **cv.findContours()**.

I. Project 1: Object counting

2. Các trường hợp cụ thể

Giải thích ý nghĩa của một số các tham số:

- Median filter kernel size: kích thước của median filter.
- Blocksize: kích thước vùng lân cận.
- Adaptive method: phương pháp được sử dụng để tìm giá trị threshold, có hai phương pháp chính được OpenCV hỗ trợ đó là:
 - `cv.ADAPTIVE_THRESH_MEAN_C`: giá trị threshold được tính bằng giá trị các pixel lân cận trừ đi một hằng số C
 - `cv.ADAPTIVE_THRESH_GAUSSIAN_C`: giá trị threshold được tính bằng tổng Gaussian có trọng số giá trị các pixel lân cận trừ đi một hằng số C
- C: một hằng số được dùng để tính toán giá trị threshold.

I. Project 1: Object counting

2. Các trường hợp cụ thể

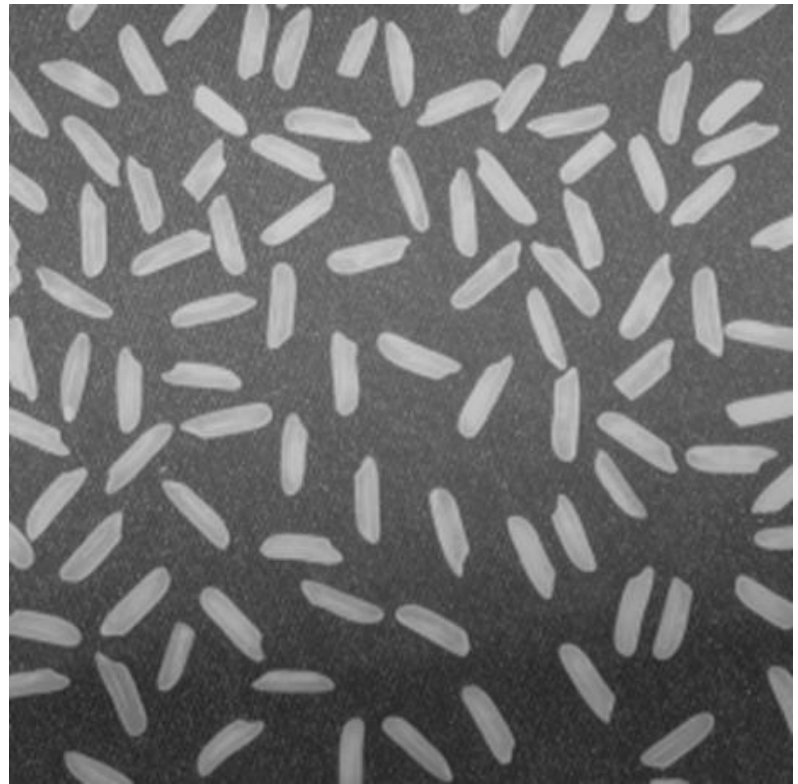
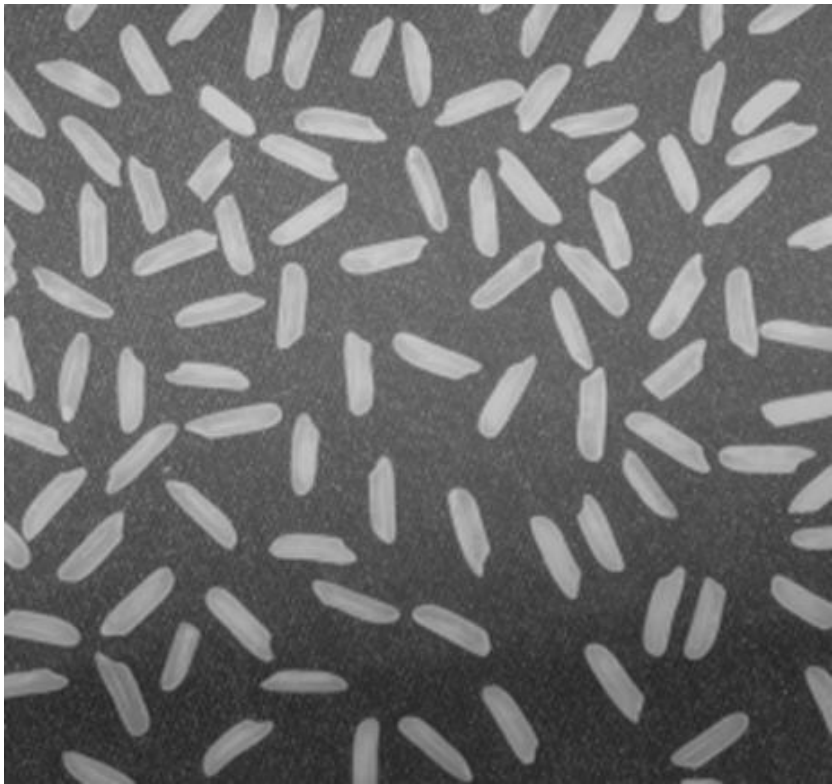
Ảnh 1, 2, 4: sử dụng bộ tham số như sau:

- Median filter: median filter kernel size 3x3.
- Adaptive Threshold:
 - Blocksize: 401
 - Adaptive method: **cv.ADAPTIVE_THRESH_GAUSSIAN_C**
 - C: 0
- Opening: opening kernel size 4x4

I. Project 1: Object counting

2. Các trường hợp cụ thể

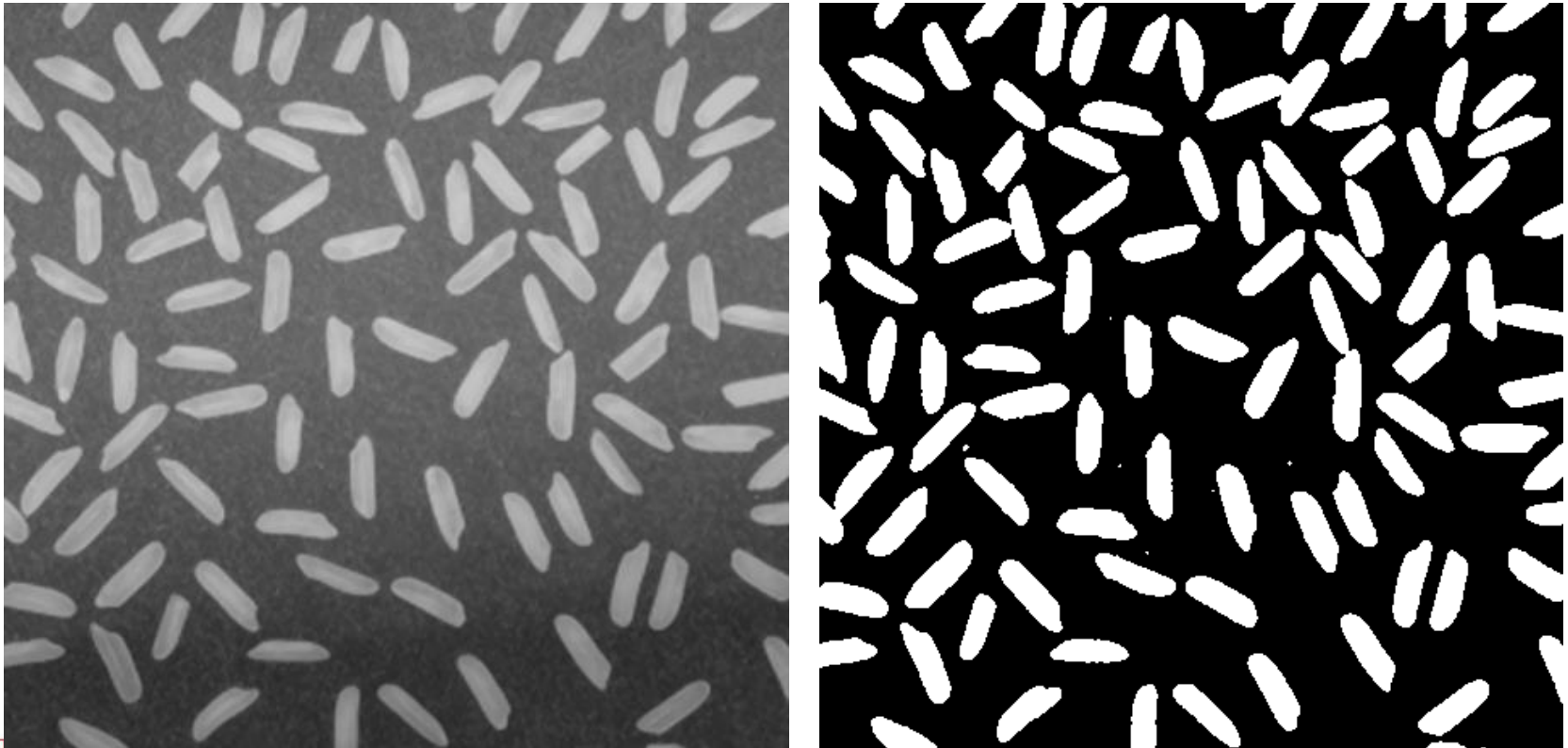
Ảnh 1: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

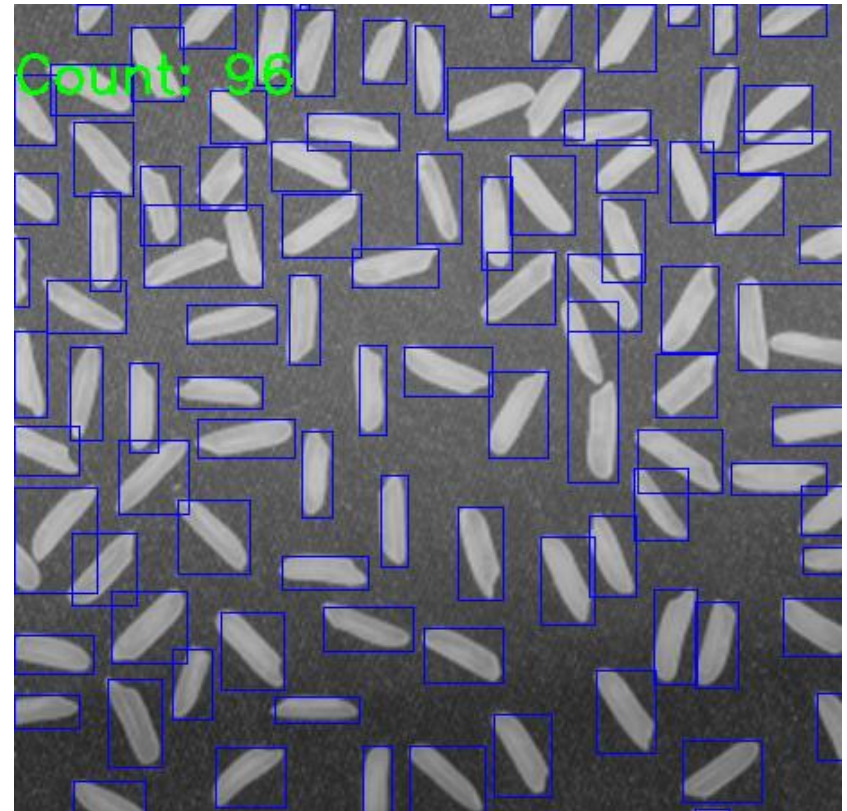
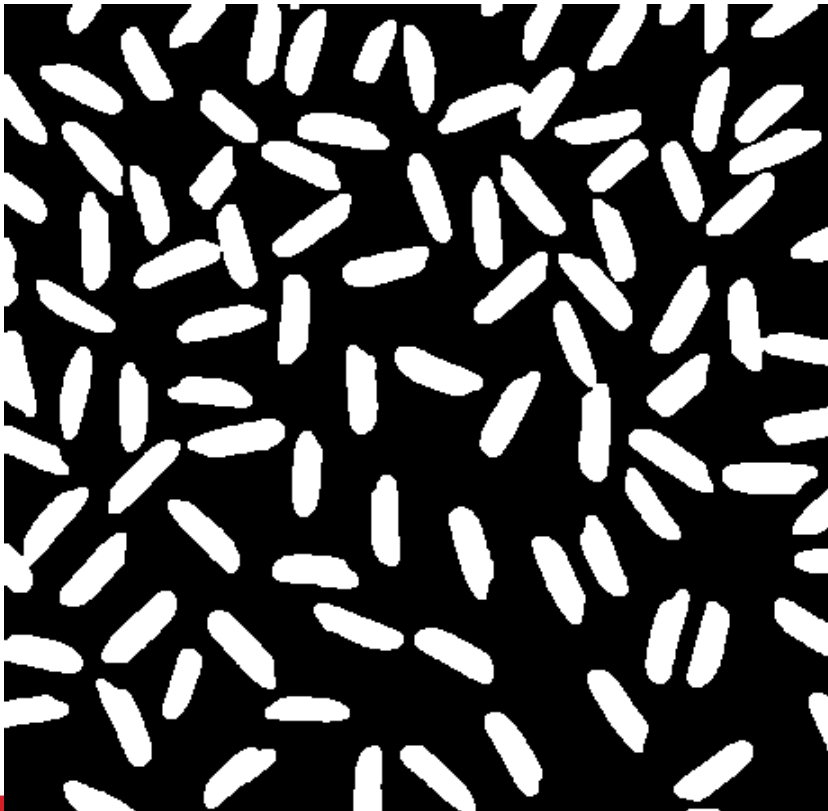
Ảnh 1: Sau khi sử dụng median filter (bên trái) và sau khi sử dụng adaptive threshold (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

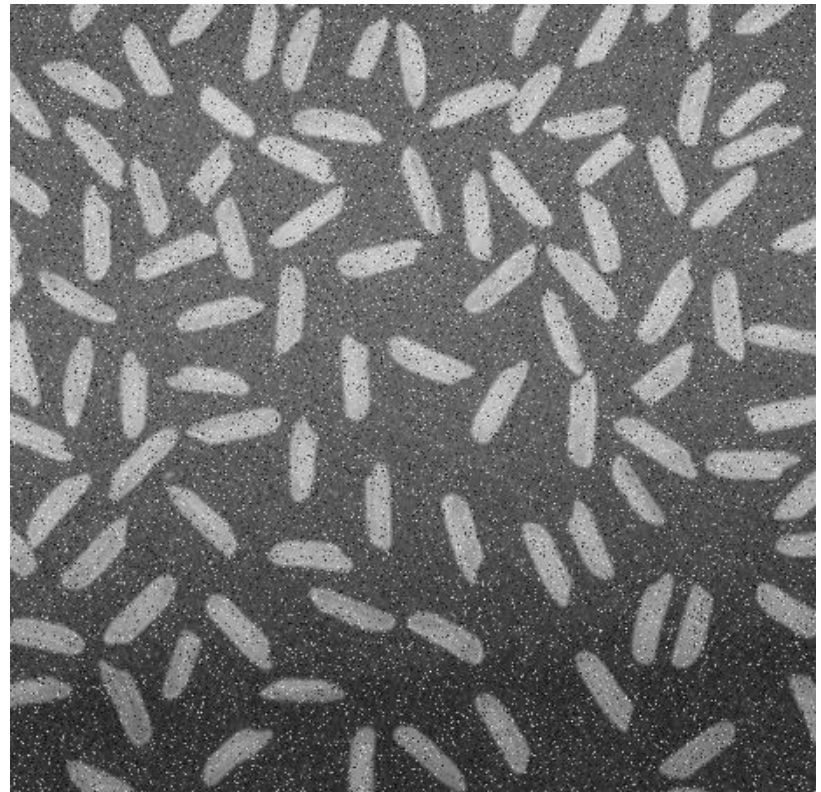
Ảnh 1: Sau khi sử dụng opening (bên trái) và kết quả cuối cùng (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

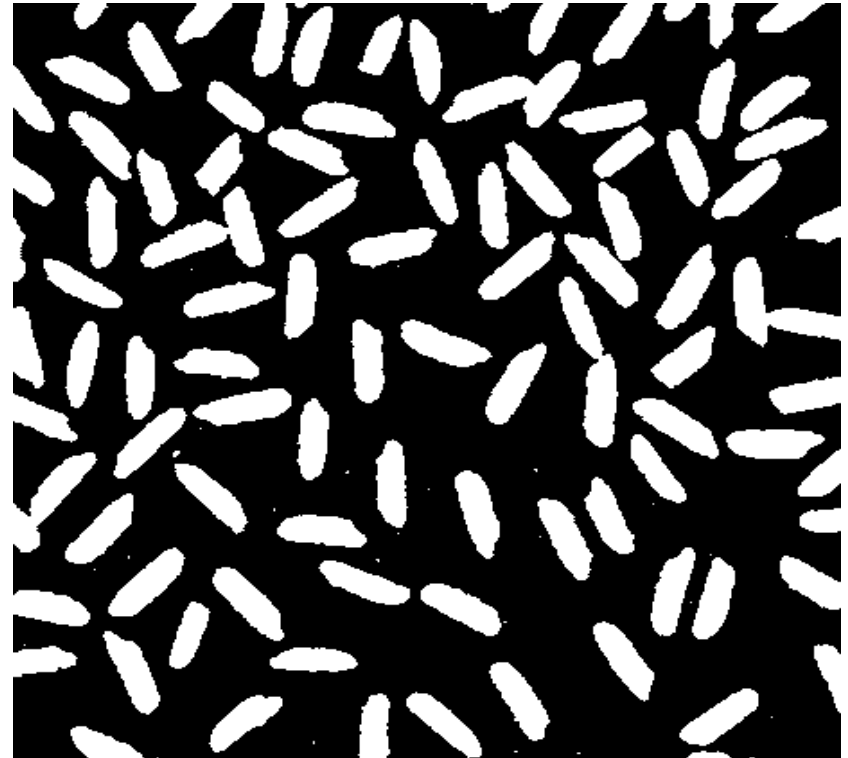
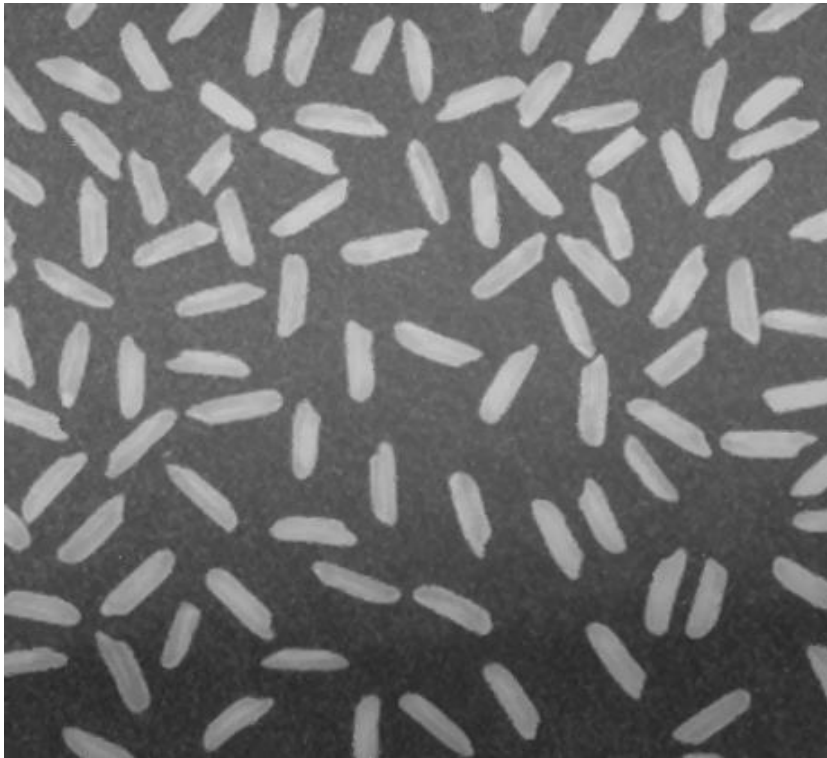
Ảnh 2: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

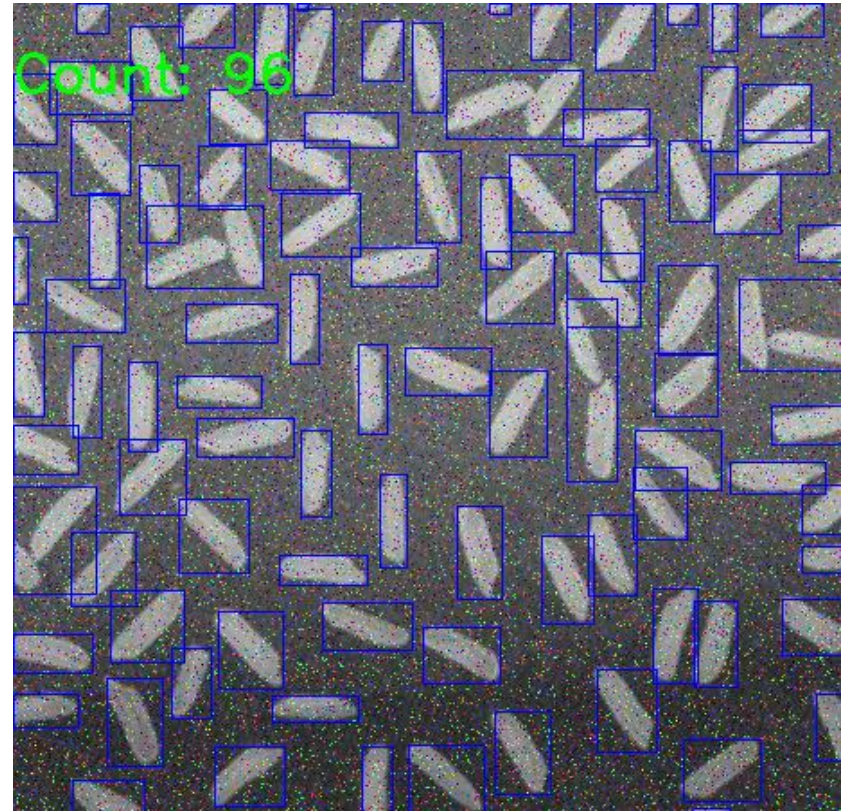
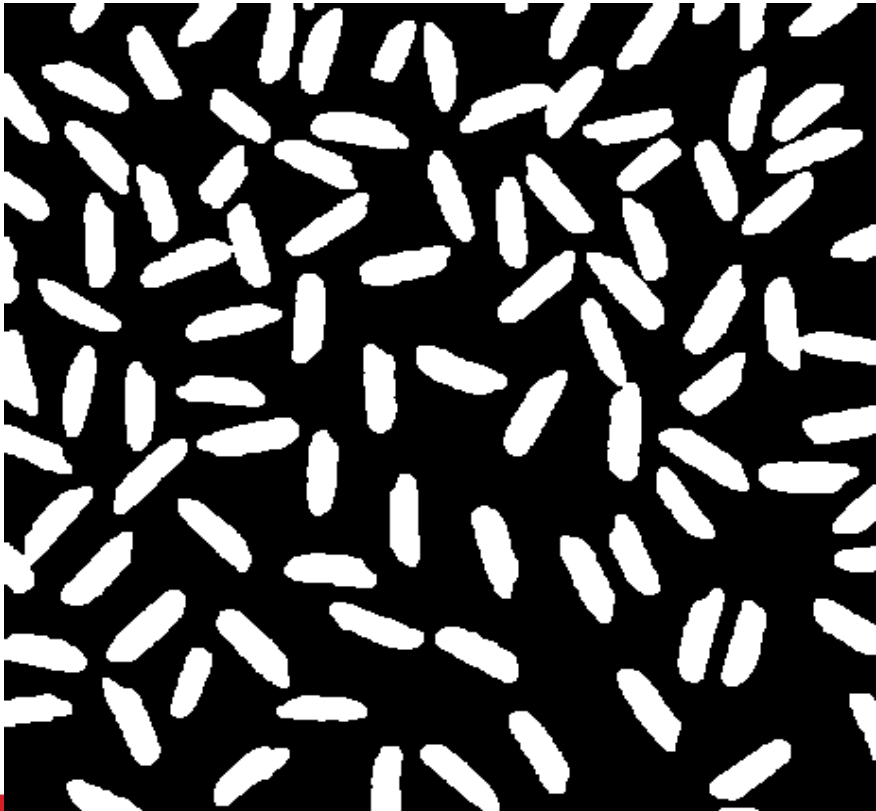
Ảnh 2: Sau khi sử dụng median filter (bên trái) và sau khi sử dụng adaptive threshold (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 2: Sau khi sử dụng opening (bên trái) và kết quả cuối cùng (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

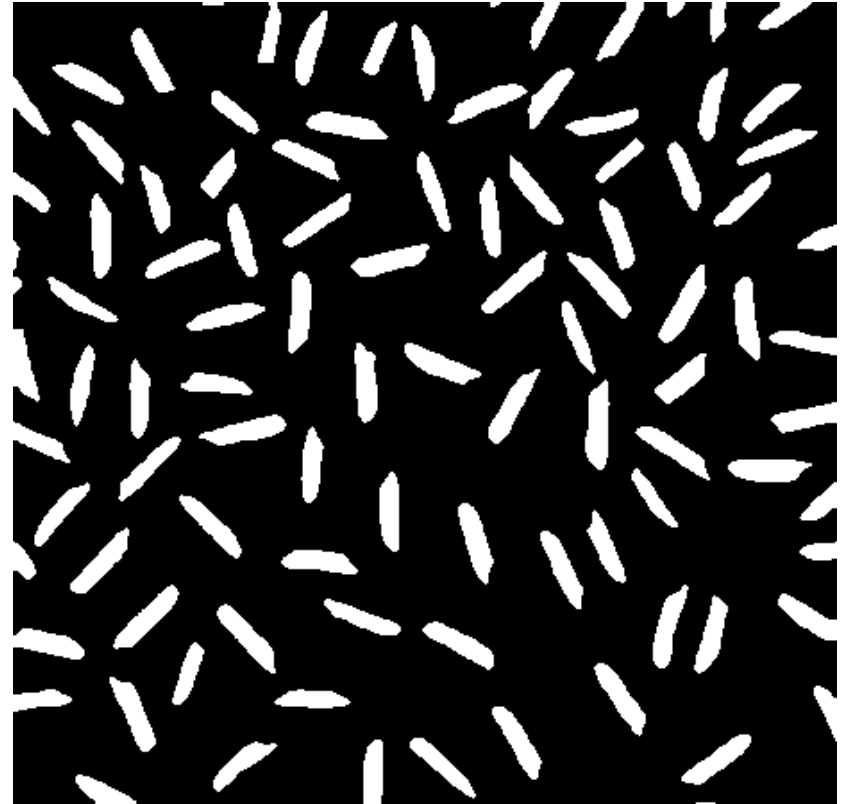
Ảnh 4: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

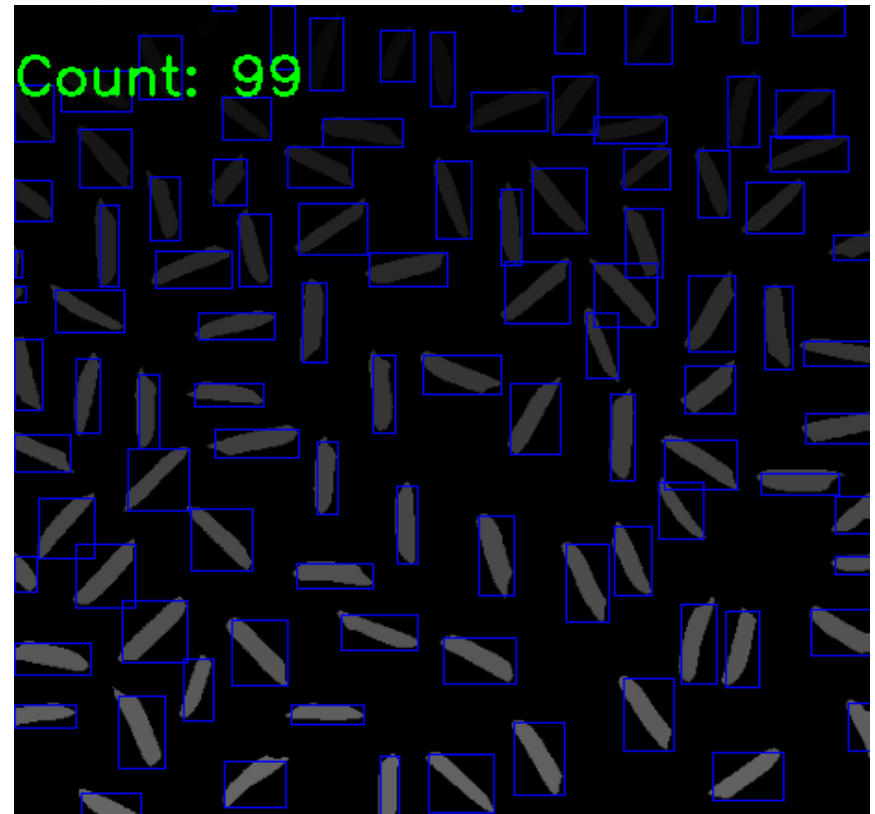
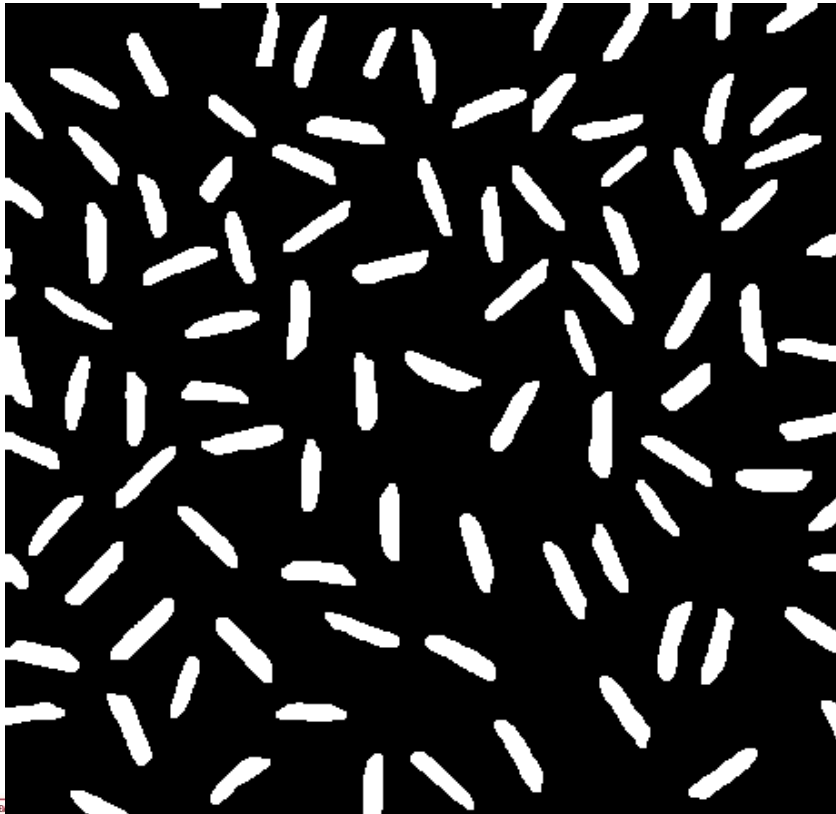
Ảnh 4: Sau khi sử dụng median filter (bên trái) và sau khi sử dụng adaptive threshold (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 4: Sau khi sử dụng opening (bên trái) và kết quả cuối cùng (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 3: ảnh có nhiều hình sin

- Chuyển ảnh sang miền tần số rồi lọc một số các tần số nhất định, rồi chuyển ngược lại về miền không gian
- Chuyển kết quả sau khi lọc về dạng unit8 và lấy nghịch đảo pixel

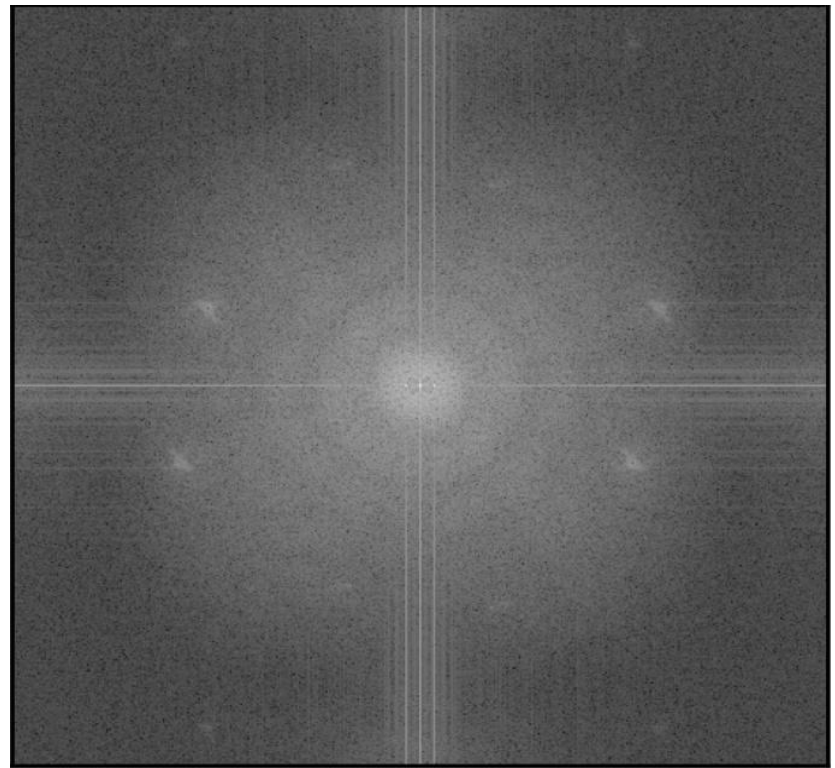
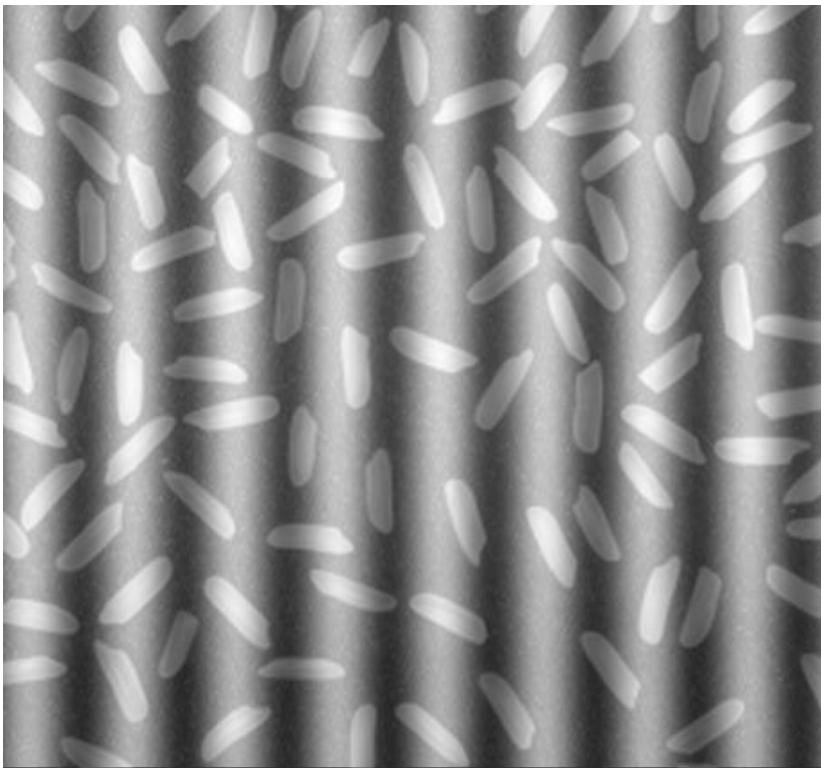
Chi tiết bộ tham số

- Adaptive Threshold:
 - Blocksize: 301
 - C: 0
 - Adaptive method: **cv.ADAPTIVE_THRESH_GAUSSIAN_C**
- Opening: closing kernel size 10 x 10

I. Project 1: Object counting

2. Các trường hợp cụ thể

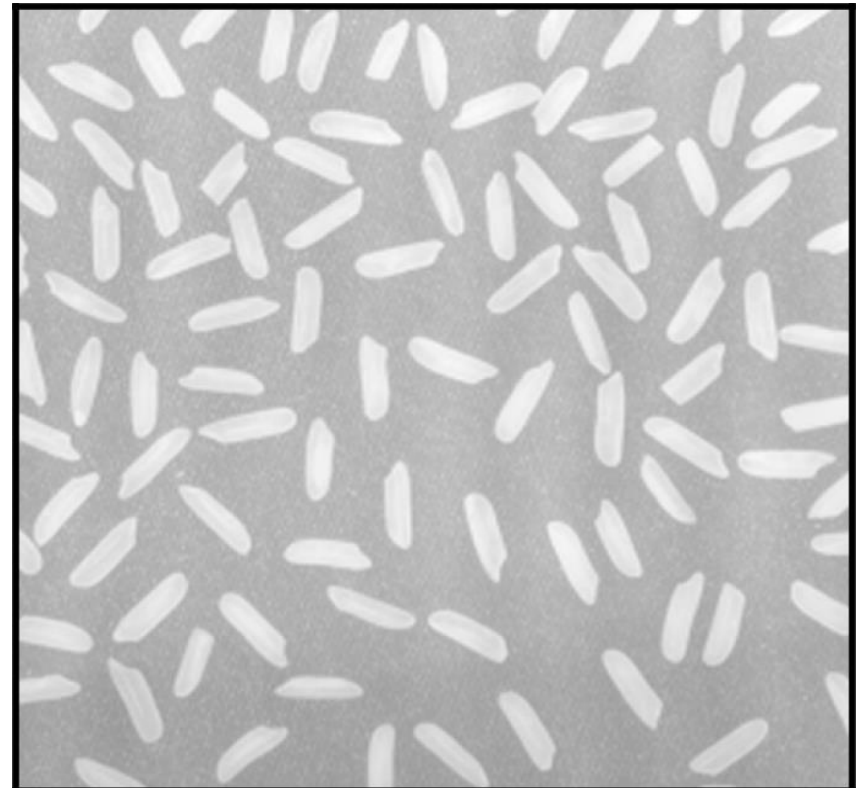
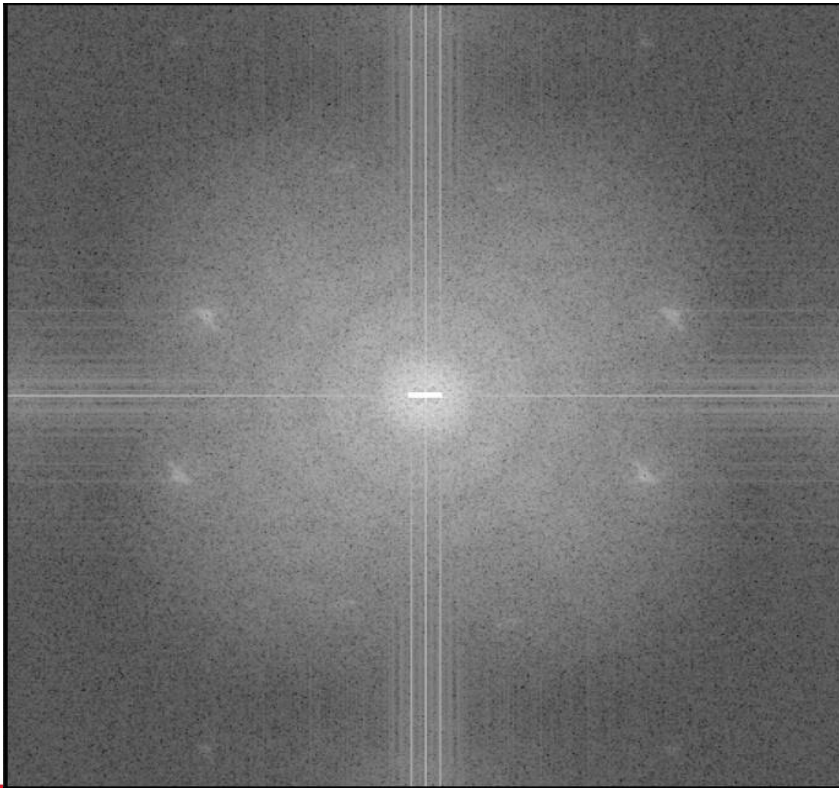
Ảnh 3: Ảnh gốc (bên trái) và sau khi chuyển sang miền tần số (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

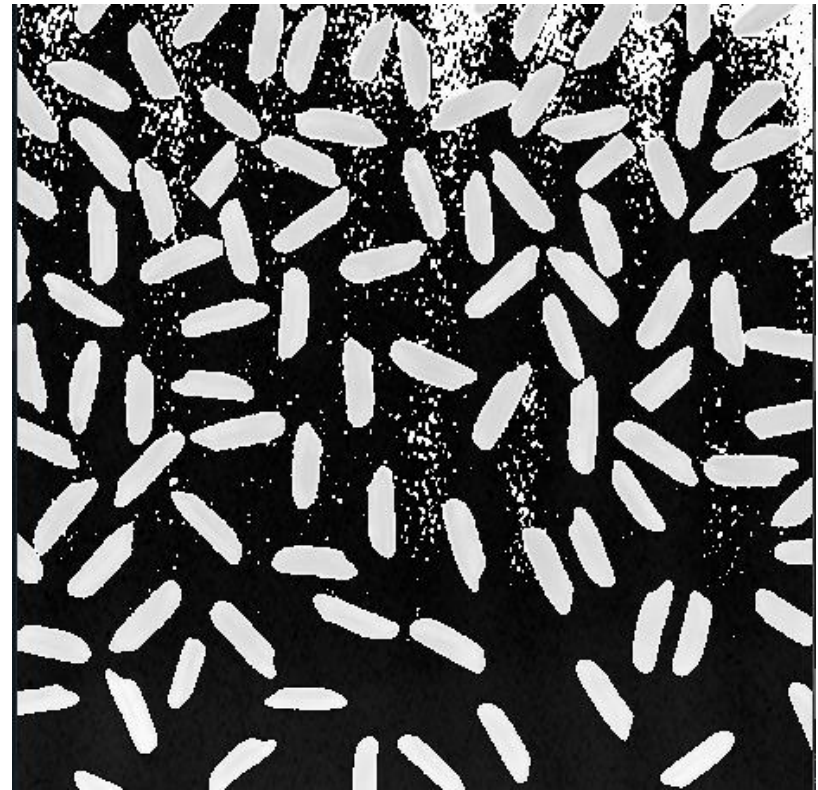
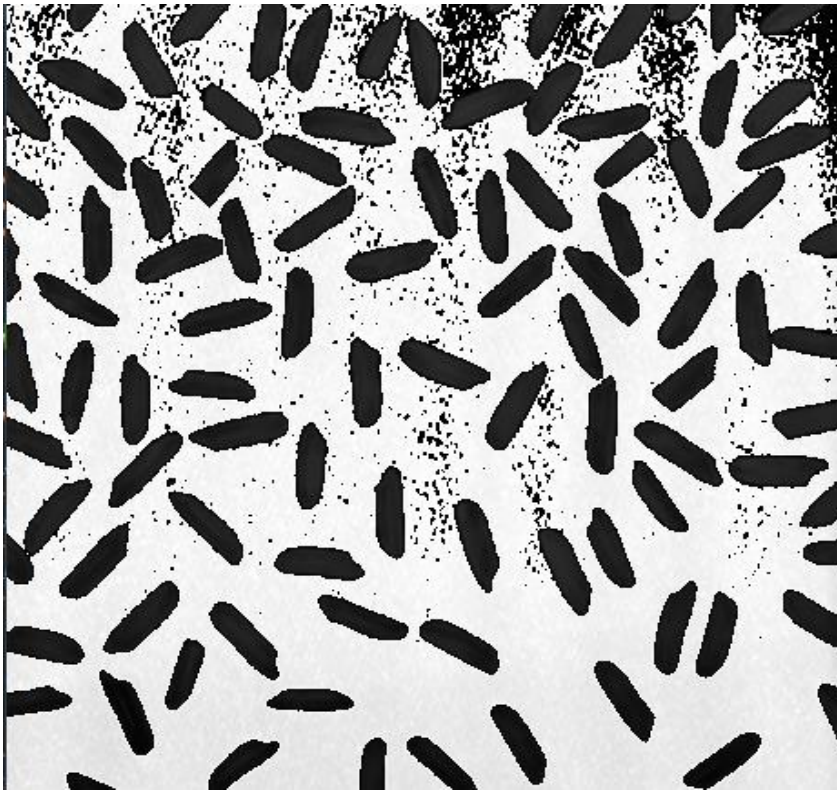
Ảnh 3: sau khi lọc nhiễu trên miền tần số (bên trái) và sau khi chuyển lại về miền không gian (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 3: Sau khi được chuyển về dạng unit8 (bên trái) và sau khi được lấy nghịch đảo pixel (bên phải)

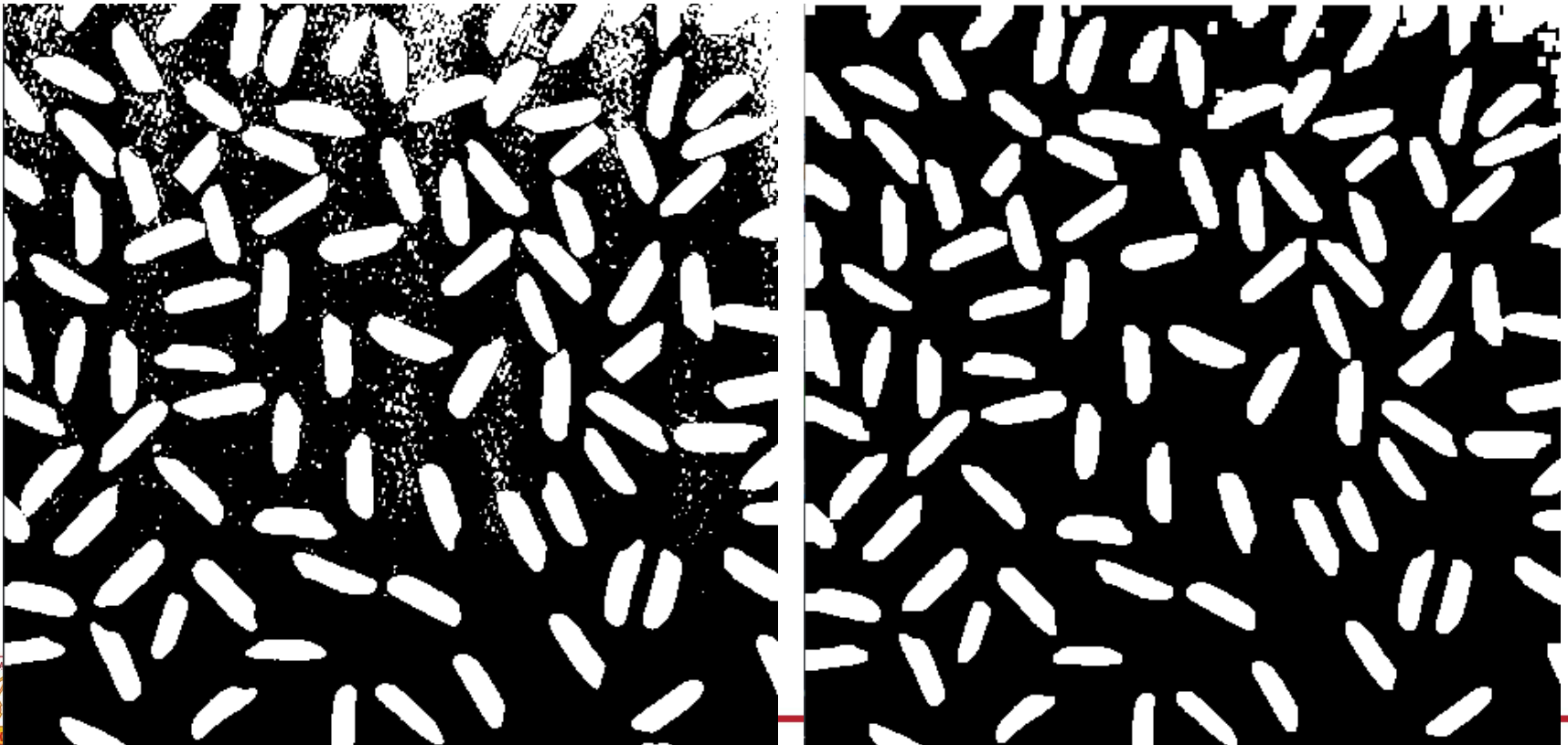


I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 3:

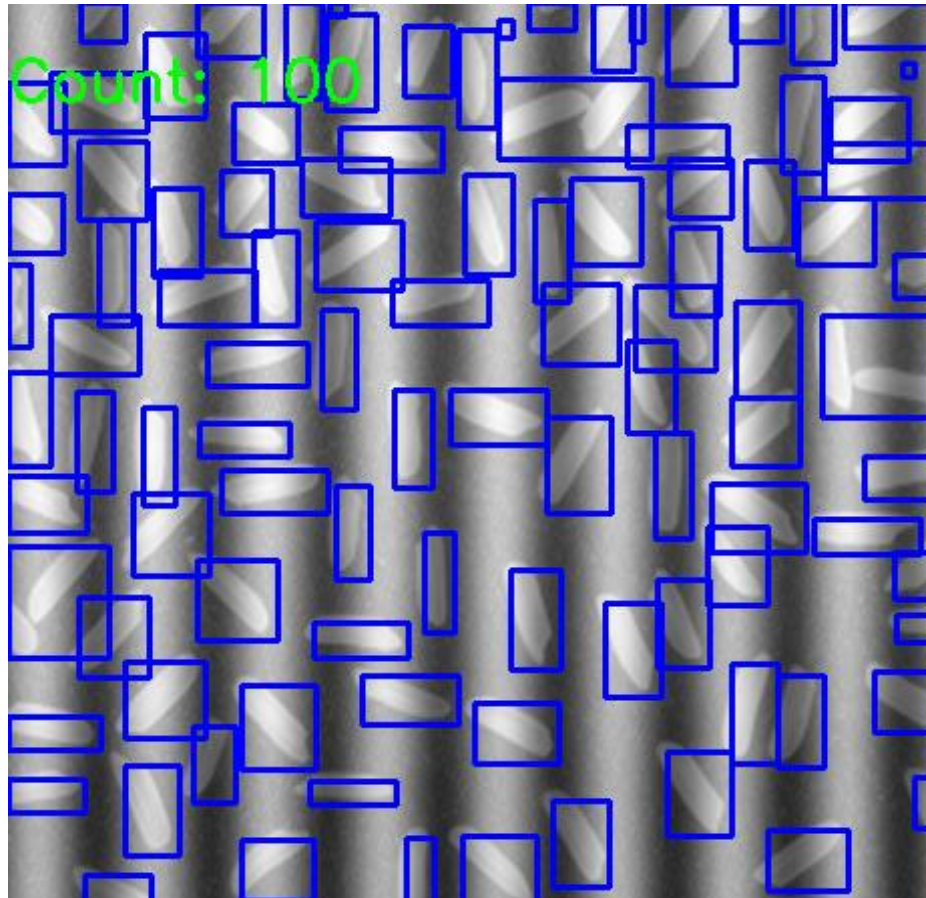
Ảnh sau dung adaptive threshold (bên trái) và sau khi sử dụng opening (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 3: Kết quả cuối cùng



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 5: không cần sử dụng sử dụng median filter, dùng closing thay cho opening

Sử dụng bộ tham số như sau:

- Median filter: median filter kernel size 3x3.
- Adaptive Threshold:
 - Blocksize: 301
 - Adaptive method: **cv.ADAPTIVE_THRESH_GAUSSIAN_C**
 - C: 20
- Closing: closing kernel size 16x16

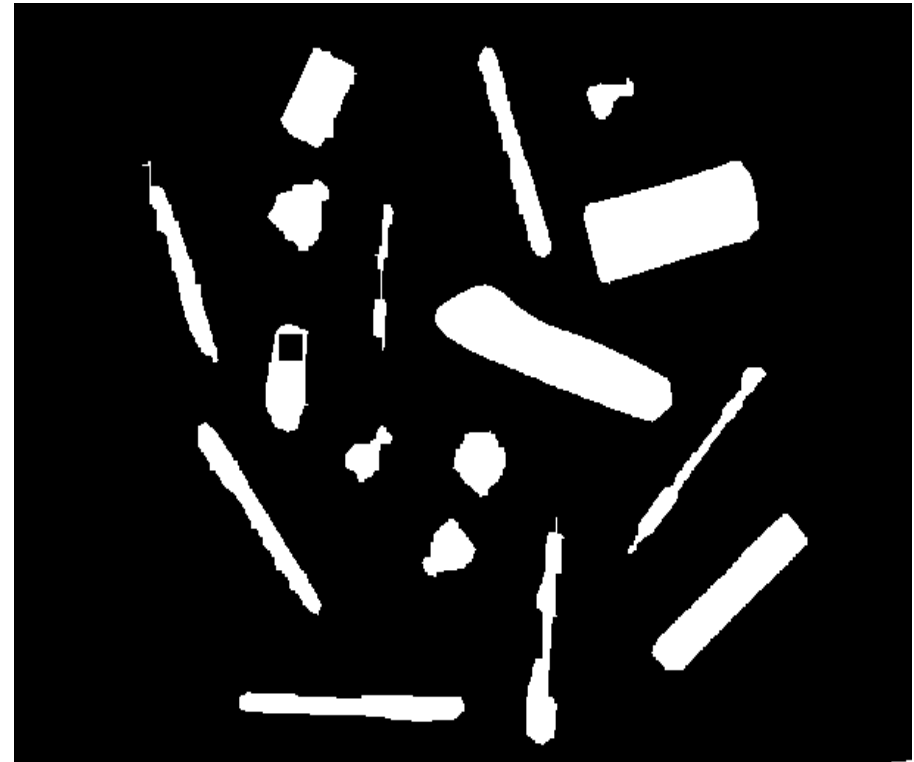
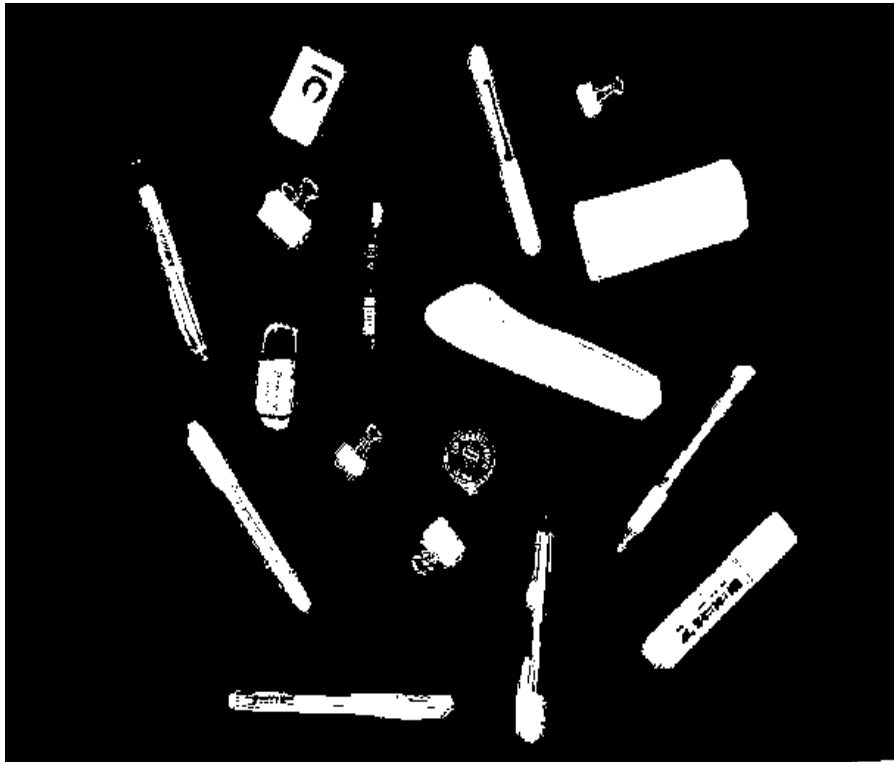
Ảnh 5: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

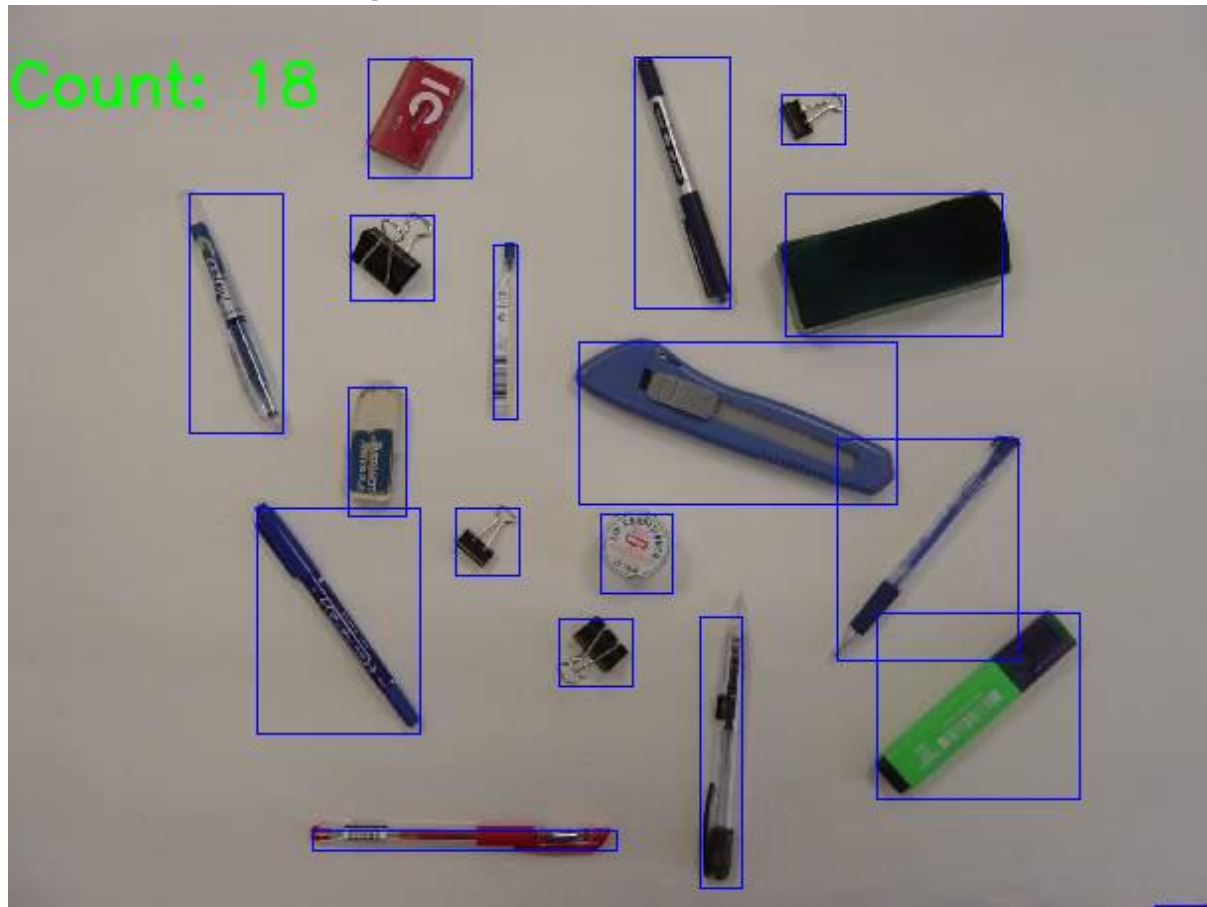
Ảnh 5: Sau khi sử dụng adaptive threshold (bên trái) và sau khi sử dụng closing (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 5: kết quả cuối cùng:



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 6, 7: sử dụng bộ tham số như sau:

- Median filter: median filter kernel size 3x3.
- Adaptive Threshold:
 - Blocksize: 401
 - Adaptive method: **cv.ADAPTIVE_THRESH_GAUSSIAN_C**
 - C: 20
- Opening: opening kernel size 16x16

I. Project 1: Object counting

2. Các trường hợp cụ thể

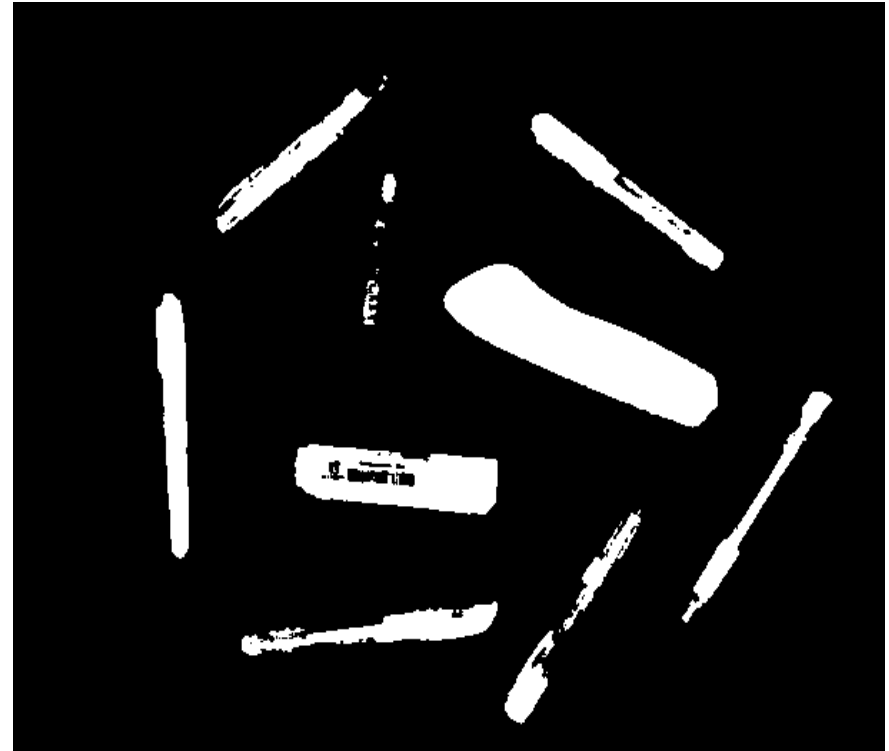
Ảnh 6: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

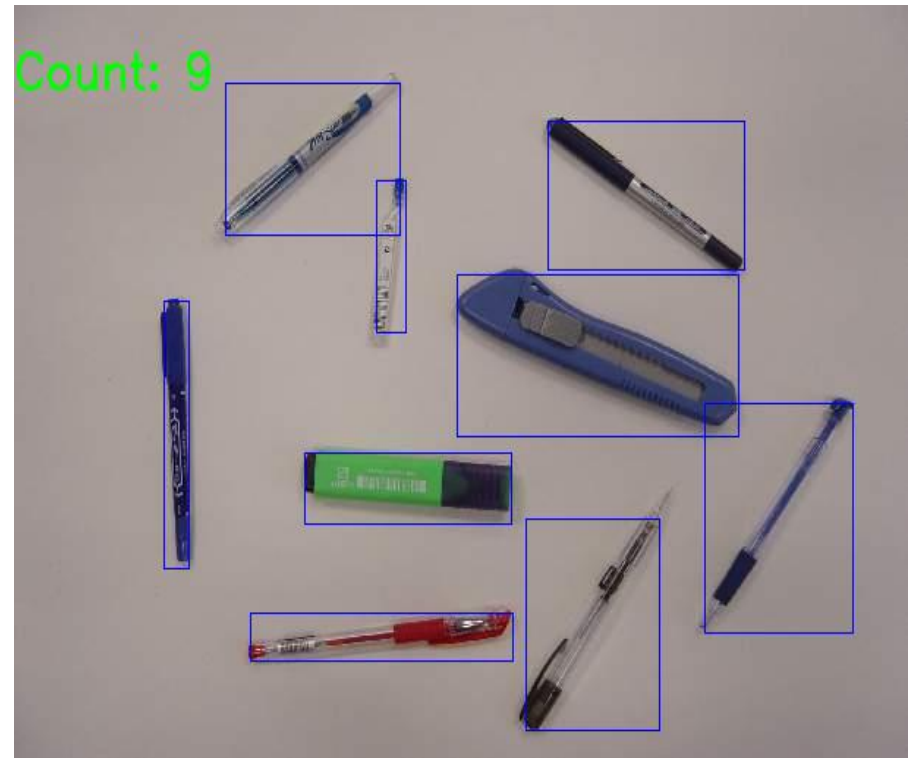
Ảnh 6: Sau khi sử dụng median filter (bên trái) và sau khi sử dụng adaptive threshold (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 6: Sau khi sử dụng closing (bên trái) và kết quả cuối cùng (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 7: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

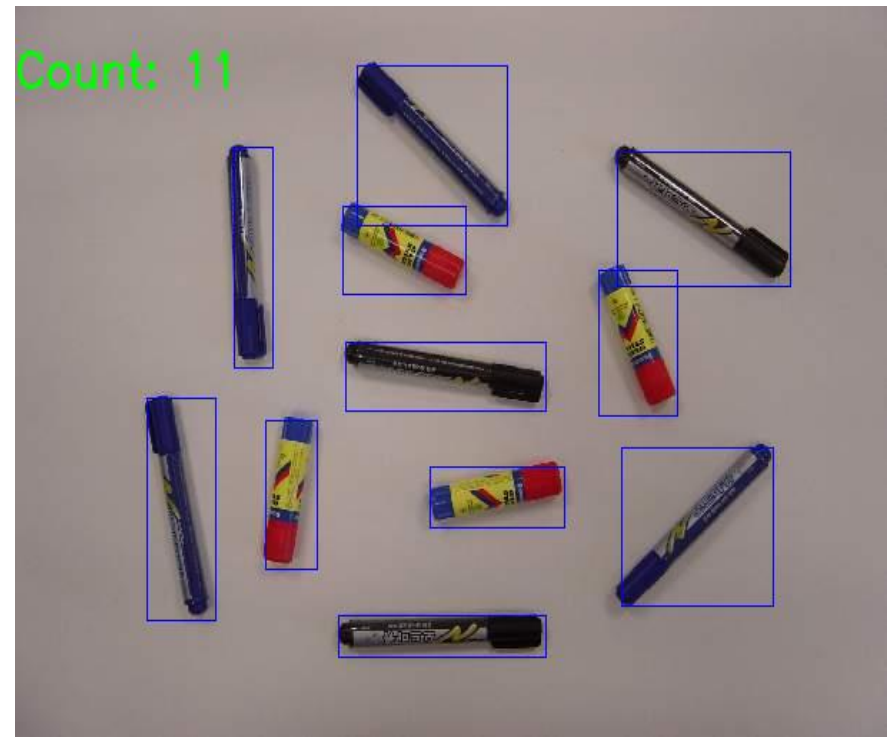
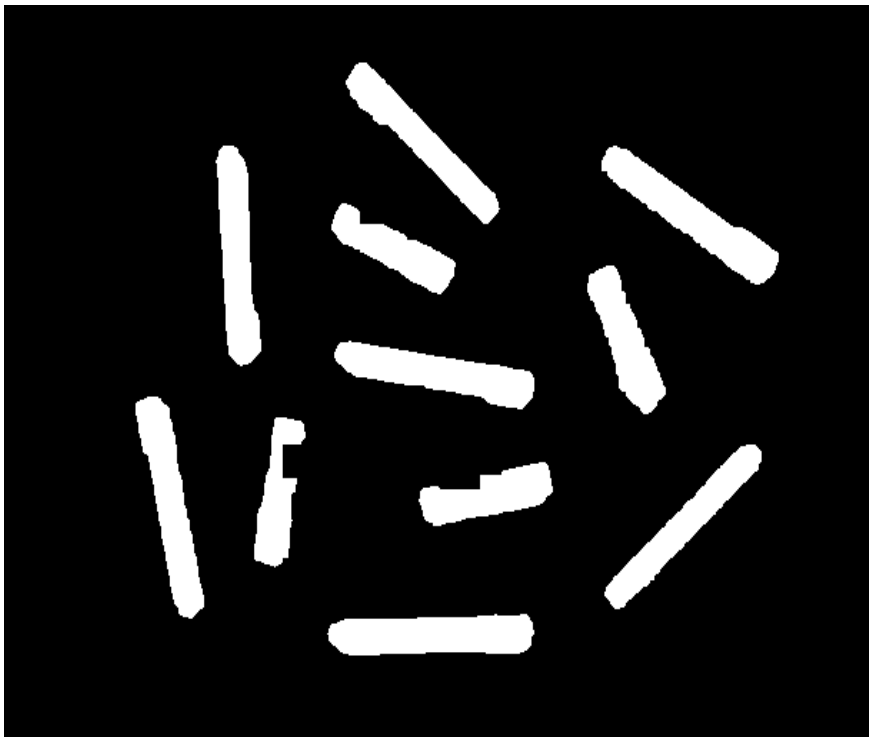
Ảnh 7: Sau khi sử dụng median filter (bên trái) và sau khi sử dụng adaptive threshold (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 7: Sau khi sử dụng closing (bên trái) và kết quả cuối cùng (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 8: không dùng median filter

Sử dụng bộ tham số như sau:

- Median filter: median filter kernel size 3x3.
- Adaptive Threshold:
 - Blocksize: 401
 - Adaptive method: **cv.ADAPTIVE_THRESH_GAUSSIAN_C**
 - C: 20
- Opening: opening kernel size 16x16

I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 6: Ảnh gốc (bên trái) và ảnh khi chuyển thành ảnh xám (bên phải)



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 6: Sau khi sử dụng adaptive threshold (bên trái) và sau khi sử dụng closing (bên phải):



I. Project 1: Object counting

2. Các trường hợp cụ thể

Ảnh 6: Kết quả cuối cùng:



II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

1. Giới thiệu về bộ dữ liệu

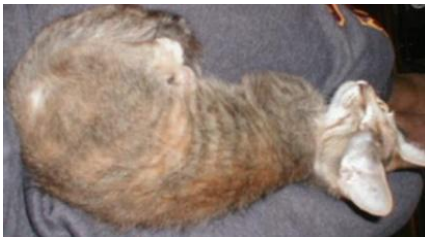
Bộ dữ liệu được sử dụng ở đây có tên là Nature Image trên Kaggle. Dữ liệu bao gồm 6899 ảnh thuộc 8 lớp khác nhau bao gồm:

- airplane: 727 ảnh.
- car: 968 ảnh.
- cat: 885 ảnh.
- dog: 702 ảnh.
- flower: 843 ảnh.
- fruit: 1000 ảnh.
- motorbike: 788 ảnh.
- person: 986 ảnh

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

1. Giới thiệu về bộ dữ liệu

Một vài mẫu dữ liệu



II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

1. Giới thiệu về bộ dữ liệu

Một vài mẫu dữ liệu



II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

1. Giới thiệu về bộ dữ liệu

Phân chia dữ liệu: chia tập dữ liệu ra làm hai tập train và test:

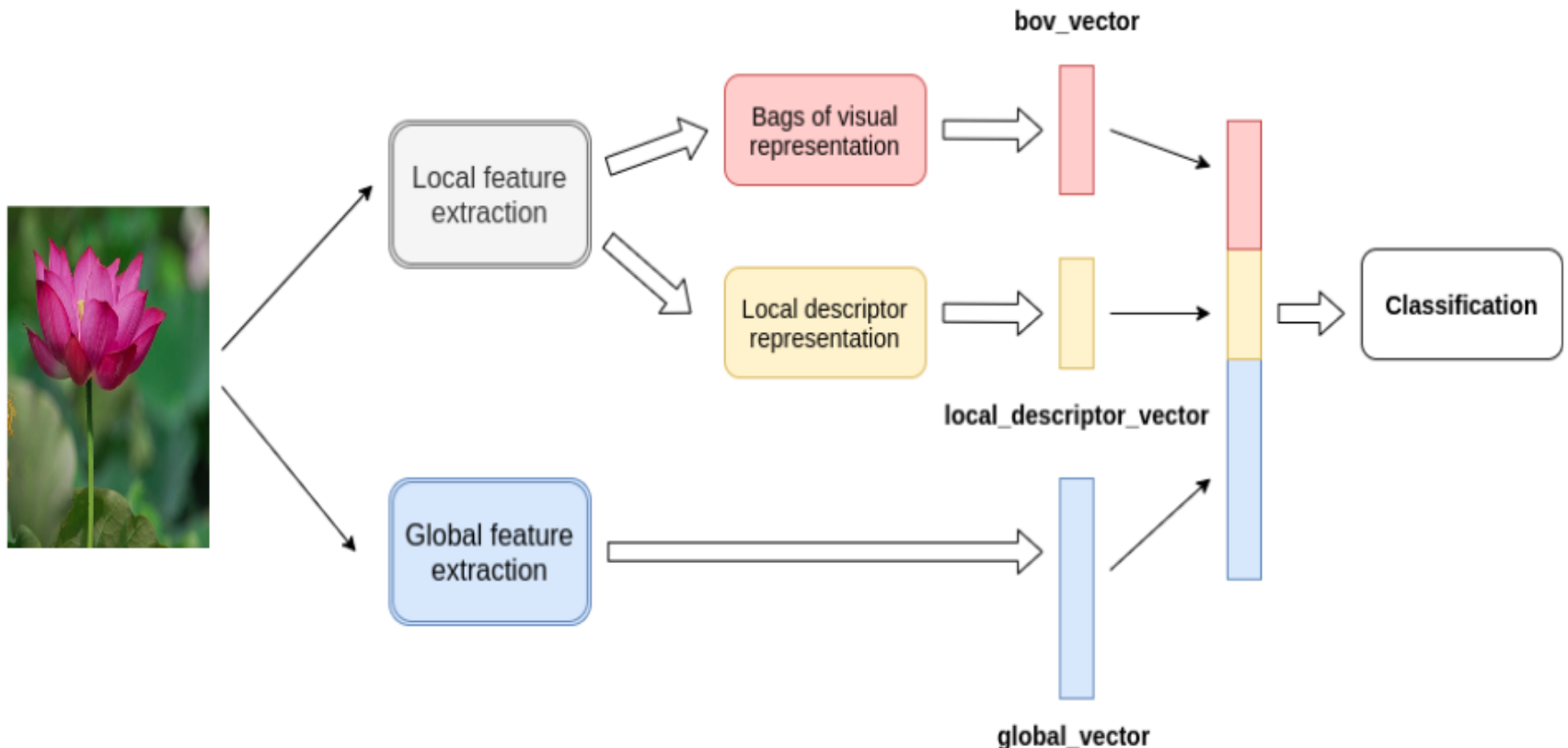
- Train: 80%
- Test: 20%

Đảm bảo tỷ lệ trên từng nhãn của ảnh

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

2. Mô hình đề xuất

Mô hình tổng quan



II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

2. Mô hình đề xuất

Bag of visual word

Mô hình phân loại dựa trên mô hình bag of visual gồm 3 bước sau:

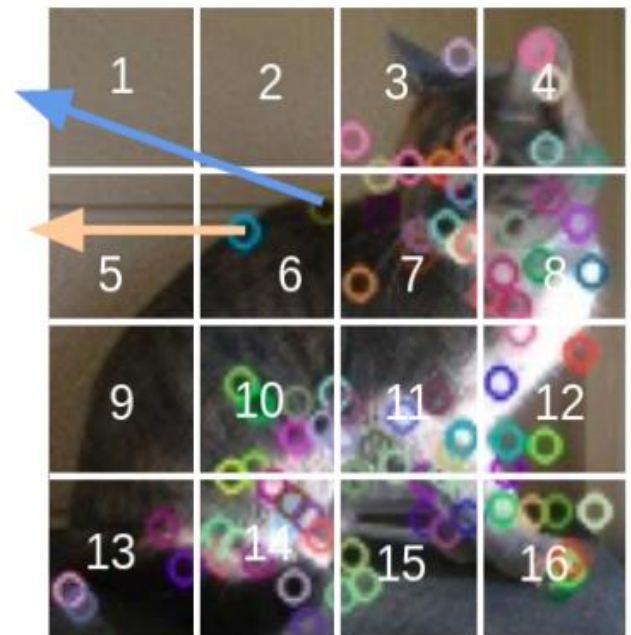
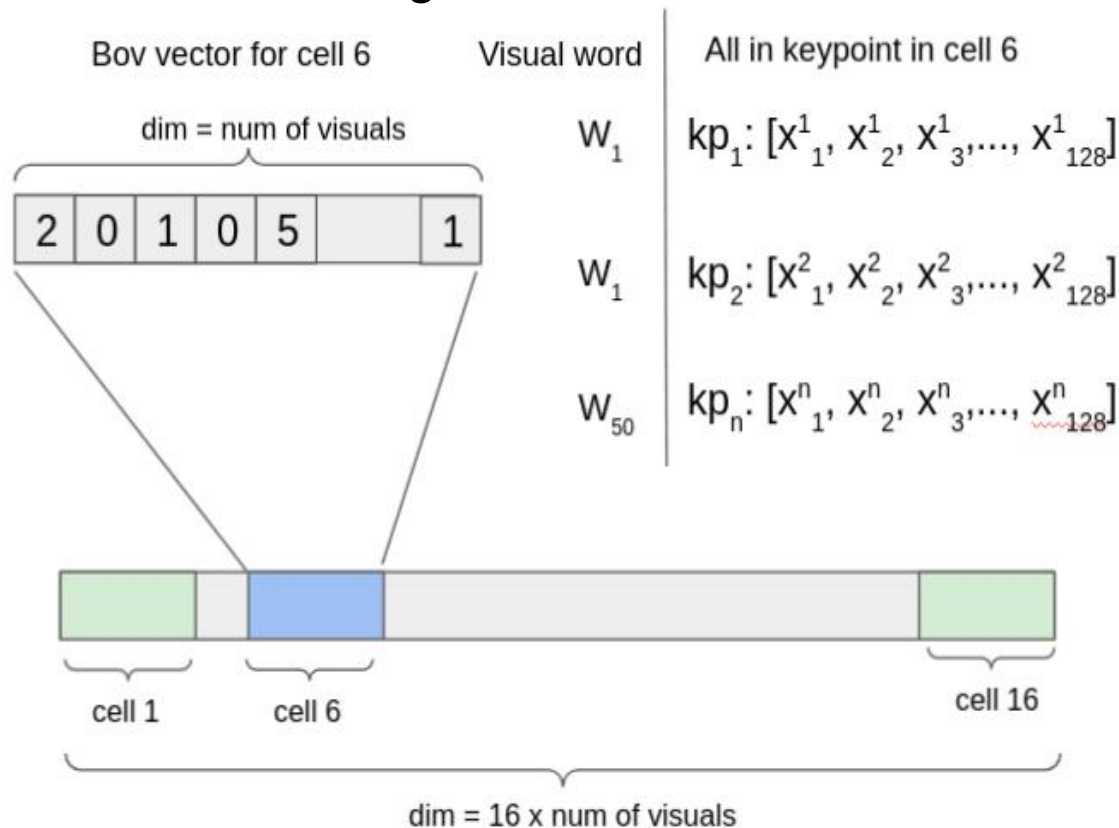
- Feature Extraction: trích xuất các descriptors là các vector đặc trưng của từng ảnh.
- Visual Word Vocabulary Construction: xây dựng bộ từ điển visual word bằng cách phân cụm các vector descriptors sử dụng thuật toán kMean.
- Image represent by bag of visual word: biểu diễn hình ảnh bằng bag of visual words dưới dạng tần suất xuất hiện của các từ.

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

2. Mô hình đề xuất

Local feature

- Local bag of visual words

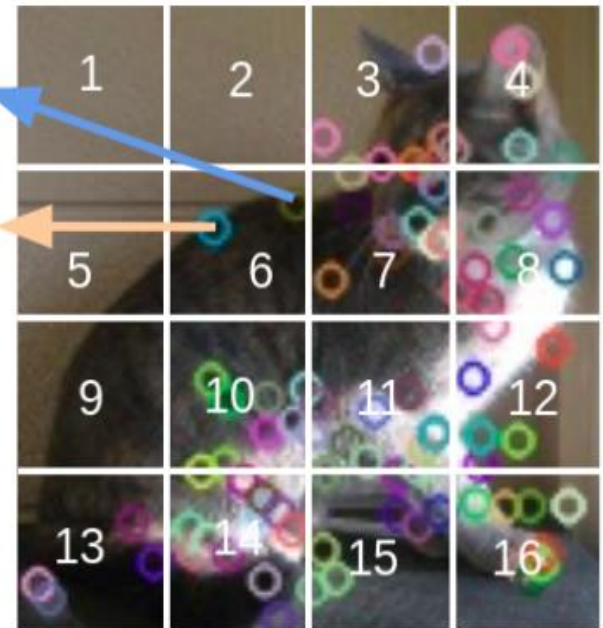
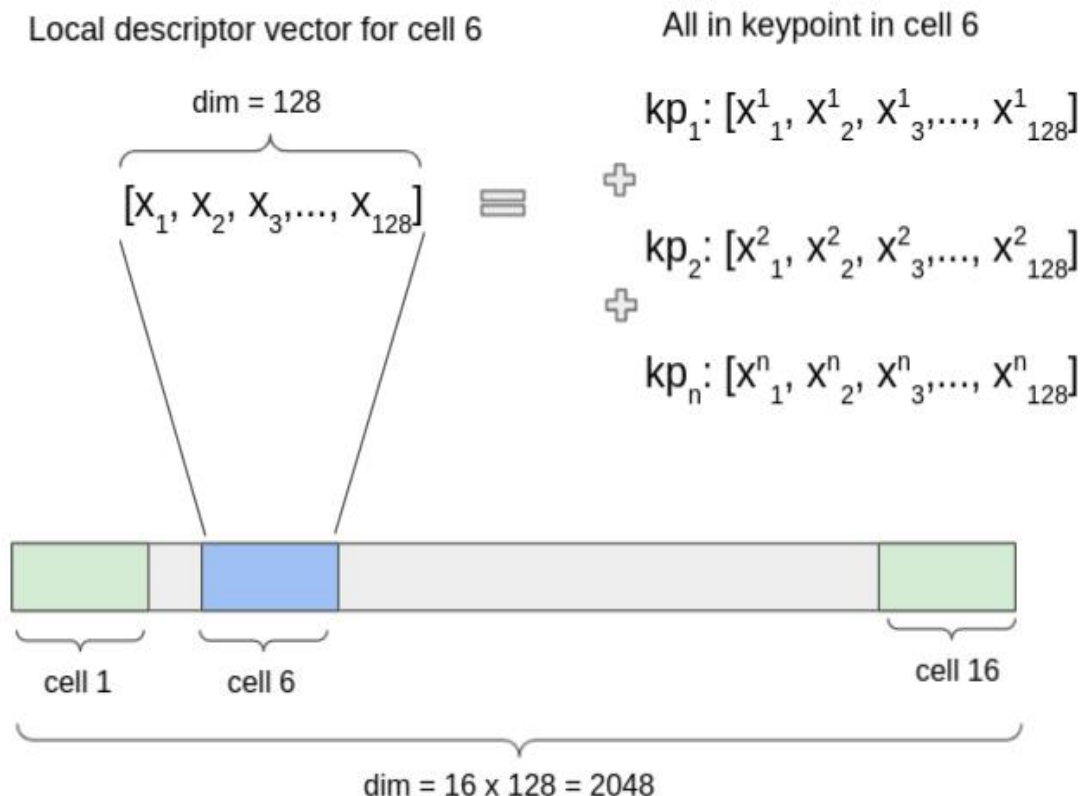


II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

2. Mô hình đề xuất

Local feature

- Local feature descriptor

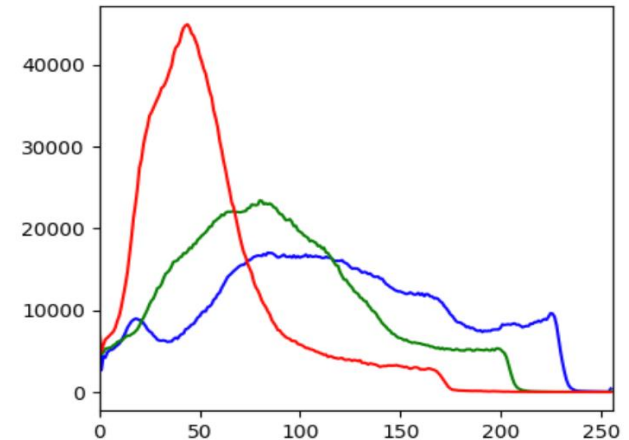


II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

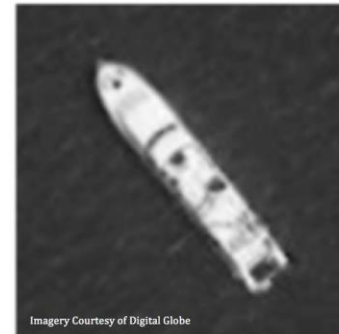
2. Mô hình đề xuất

Global feature

- Histogram cho 3 chanel RGB của ảnh



- Histogram of oriented gradients – HoG



II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

3. Kết quả thực nghiệm

Mô hình BoW cơ bản

Num of visuals	Accuracy	Precision	Recall	F1-Score
50	0.6377	0.6258	0.6247	0.6245
100	0.6876	0.6773	0.6756	0.6762
200	0.7187	0.7069	0.7064	0.7062
300	0.7237	0.7130	0.7125	0.7125
400	0.7483	0.7391	0.7402	0.7393
500	0.7324	0.7232	0.7226	0.7226

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

3. Kết quả thực nghiệm

Mô hình local bags of word và local feature descriptor ($n_{\text{visual}} = 400$)

Model	Accuracy	Precision	Recall	F1-Score
BoW	0.7483	0.7391	0.7402	0.7393
BoW + local bag of word	0.8055	0.8055	0.8027	0.8035
BoW + local feature descriptor	0.8040	0.8040	0.7924	0.7924
BoW + local bag of word + local feature descriptor	0.8011	0.7897	0.7901	0.7897

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

3. Kết quả thực nghiệm

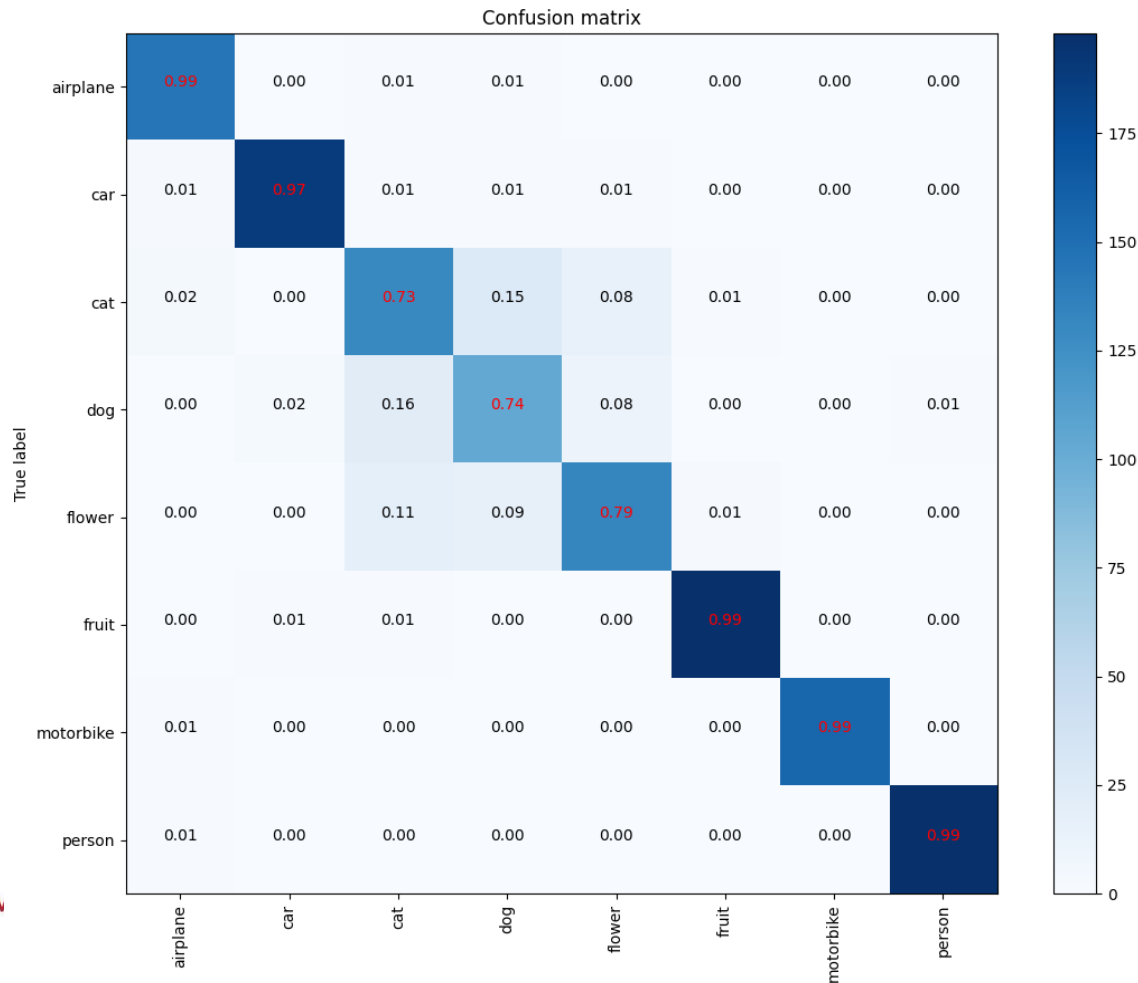
Mô hình BoW kết hợp với global feature và local feature

Model	Accuracy	Precision	Recall	F1-Score
BoW	0.7483	0.7391	0.7402	0.7393
BoW + local feature descriptor + histograms	0.8365	0.8292	0.8292	0.8277
BoW + local feature descriptor + HoG	0.8886	0.8803	0.8798	0.8791
BoW + local feature descriptor + histograms + HoG	0.8958	0.8860	0.8871	0.8864
BoW + local BoW + histograms	0.8459	0.8408	0.8369	0.8385
BoW + local BoW + HoG	0.9009	0.8928	0.8935	0.8931
BoW + local BoW + histograms + HoG	0.9052	0.8984	0.8997	0.8988
Basic CNN	0.9178	0.9101	0.9106	0.9102

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

3. Kết quả thực nghiệm

Kết quả chi tiết cho mô hình BoW + local BoW + histograms + HoG:



II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

4. Kết luận

- Các mô hình sử dụng cả global feature và local feature cho kết quả tốt nhất.
- Các trường cat, dog và flower cho kết quả thấp hơn so với các trường khác => Sử dụng thêm dữ liệu để huấn luyện thêm cho 3 trường này
- Kết quả khi sử dụng local bag of word có cao hơn so với việc sử dụng local feature descriptor.

II. Project II: nhận dạng đối tượng sử dụng mô hình BoW

4. Kết luận

- Tương lai sẽ huấn luyện mô hình sử dụng cả local BoW, local feature descriptor, histogram và HoG.
- Dùng một số các local feature extractor khác như SURF, ORB, .. Và các global feature khác như invariant moments, PHOG, Co-HOG,...
- Thử nghiệm huấn luyện trên các tập dữ liệu khác lớn hơn như CIFAR10, COIL100, ...

III. Tài liệu tham khảo

- Slide môn học Thị giác máy tính – IT5409
- LEARN OPENCV in 3 HOURS with Python | Including 3xProjects | Computer Vision:
<https://www.youtube.com/watch?v=WQeoO7MI0Bs&t=5849s>
- Nature Image Dataset:
<https://www.kaggle.com/prasunroy/natural-images>
- Bag of Visual Words: <https://github.com/gurkandemir/Bag-of-Visual-Words>



25 YEARS ANNIVERSARY
SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

**Thank you
for your
attentions!**



soict.hust.edu.vn/



fb.com/groups/soict

