Phần I

Classical Computer Vision – Rice Grain Counting

0.1 Project Overview

The primary objective of this project is to develop an automated system for counting rice grains from digital images. This task is foundational for agricultural automation and quality control. While counting distinct objects may seem trivial, real-world imaging conditions introduce several technical hurdles that must be addressed through a structured image processing pipeline.

0.1.1 Environmental Challenges

The system is designed to handle four specific image-based problems:

- **Uneven Illumination:** Shadows and vignetting that make global thresholding impossible.
- **Sensor Noise:** High-frequency "salt-and-pepper" noise that can be mistaken for grains.
- **Patterned Backgrounds:** Structural interference (wavy patterns) that mimics grain intensity.
- **Grain Clustering:** Multiple grains touching or overlapping, requiring advanced separation techniques.

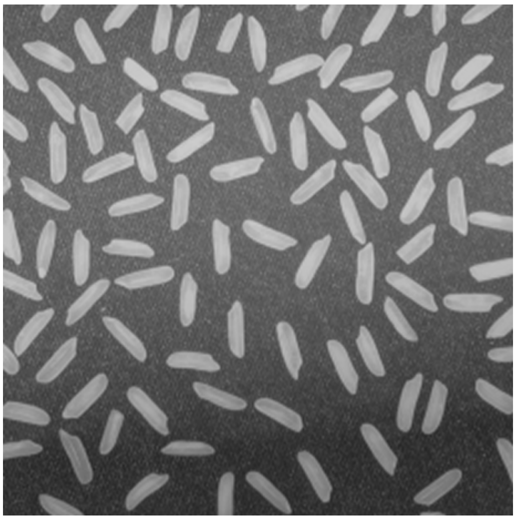
0.1.2 Proposed Processing Workflow

To return an accurate count, the solution follows a sequential logic:

1. **Preprocessing:** Applying Top-Hat transforms to fix uneven lighting and Median Blurs to remove noise.
2. **Background Neutralization:** Using directional kernels and image subtraction to isolate foreground objects from complex backgrounds.
3. **Segmentation:** Utilizing Otsu's Thresholding to generate a binary mask of the rice grains.
4. **Morphological Refinement:** Applying Morphological Opening to clean artifacts.
5. **Object Separation:** Implementing the Watershed Algorithm to identify and count individual grains within clusters.

0.2 Case Study 1: Illumination and Noise (Pic 1 & 2)

Images 1 and 2 represent the most common capture issues: varying light levels across the surface and granular noise.



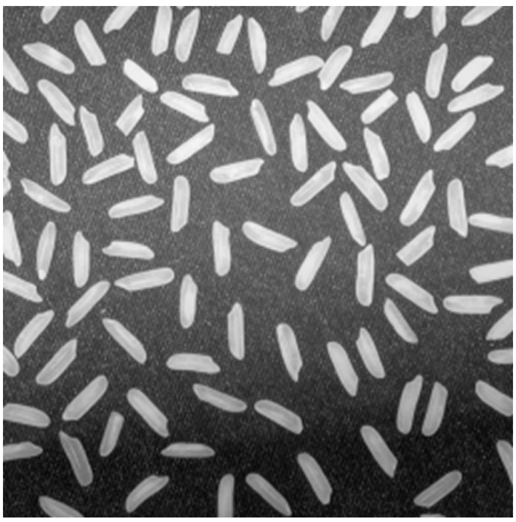
Hình 0.1: Original Image 1 (Uneven Light)

Hình 0.2: Original Image 2 (High Noise)

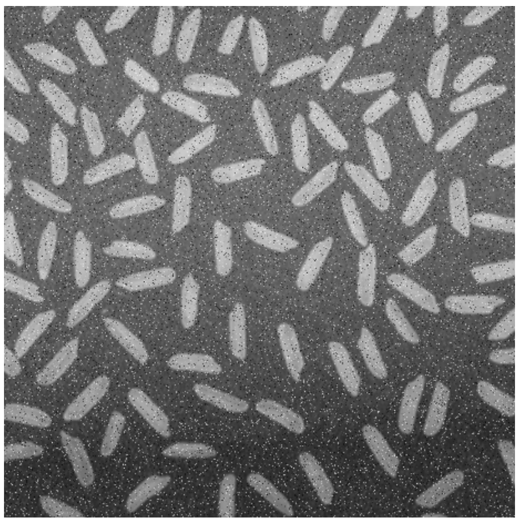
0.2.1 Implementation Details

For these cases, a specific preprocessing configuration was used:

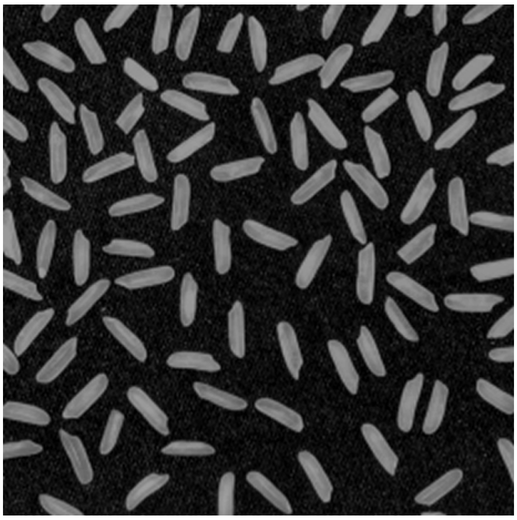
- **Top-Hat Transform:** A kernel size of 40×40 was applied to extract bright objects from the dark, uneven background.
- **Median Blur:** A kernel size of 5 was used to suppress noise while maintaining the sharp edges of the grains.
- **Thresholding:** Otsu's method was applied to verify rice pixels versus the background.



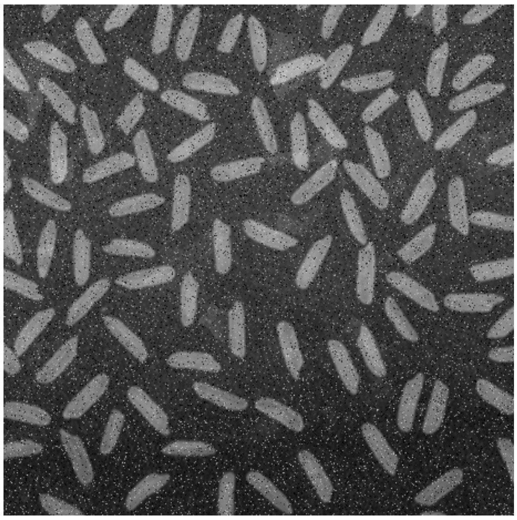
Hình 0.3: Image 1 Grayscale



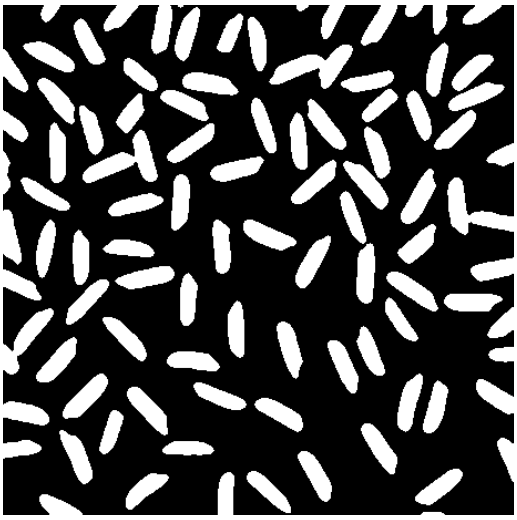
Hình 0.4: Image 2 Grayscale



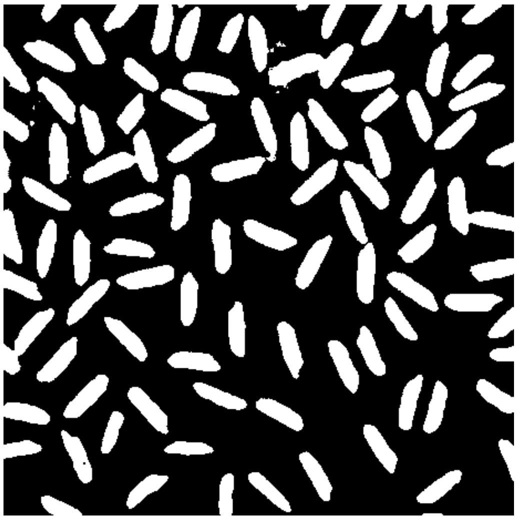
Hình 0.5: Image 1 Top-Hat



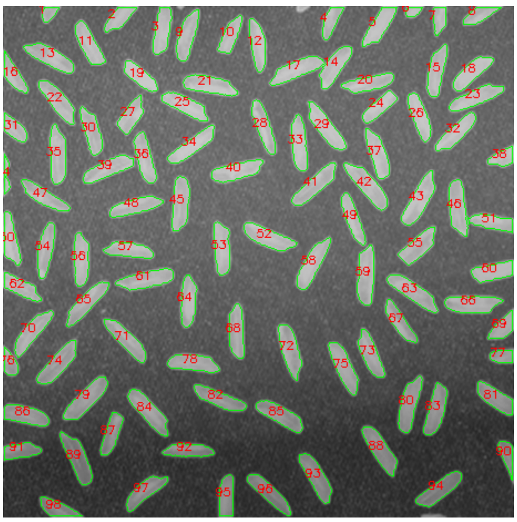
Hình 0.6: Image 2 Top-Hat



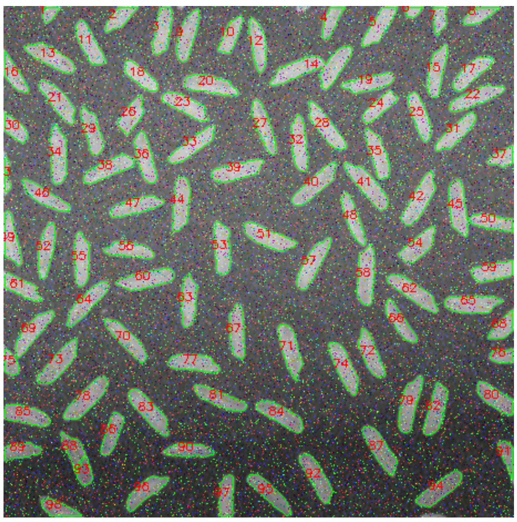
Hình 0.7: Image 1 Threshold



Hình 0.8: Image 2 Threshold



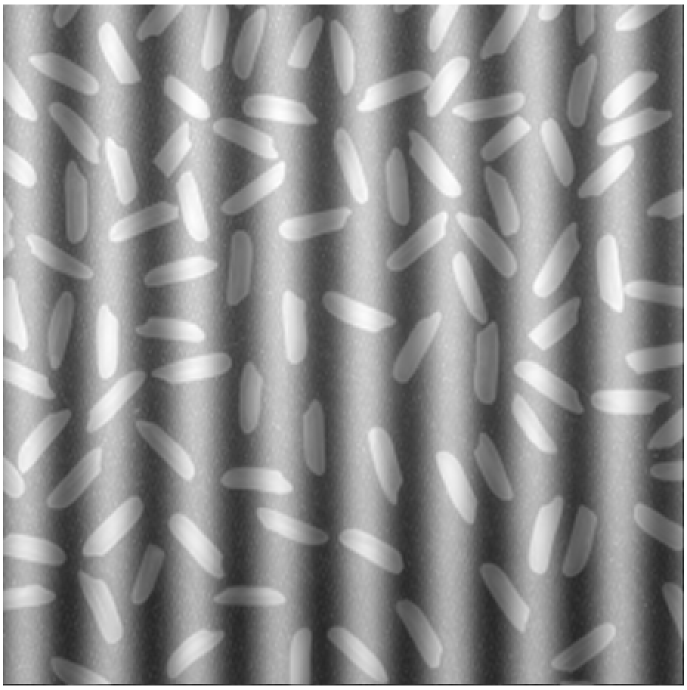
Hình 0.9: Image 1 Results



Hình 0.10: Image 2 Results

0.3 Suppressing Patterned Backgrounds (Pic 3)

Image 3 presents a specialized challenge where rice grains are placed on a surface with high-frequency vertical wavy patterns. The peaks of these waves possess pixel intensities nearly identical to the rice grains, rendering global thresholding ineffective as it would include the background structure in the final count.



Hình 0.11: Original Image 3 with Vertical Wave Interference

0.3.1 Pattern Isolation and Subtraction Logic

To neutralize the background, the system employs a directional filtering approach followed by image arithmetic:

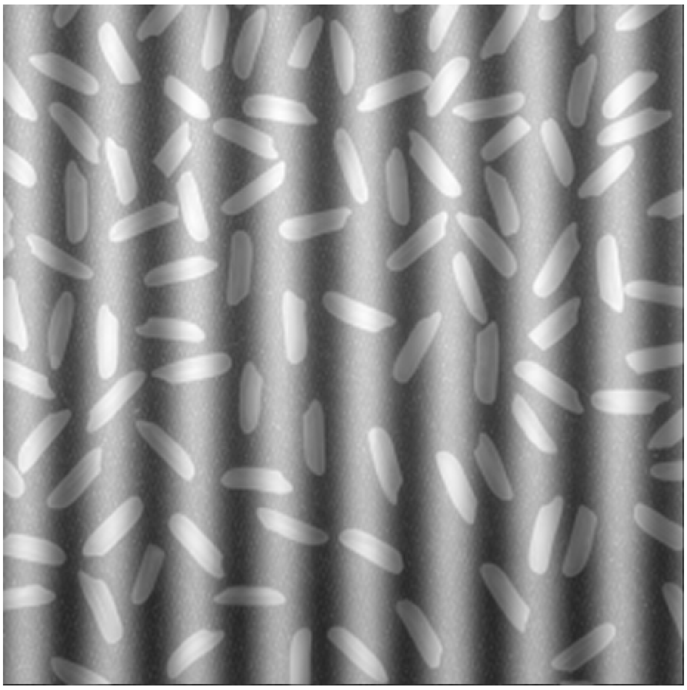
- **Vertical Kernel Filtering:** A specialized 1×80 vertical kernel is applied to the image. This operation is designed to specifically isolate the vertical structural components of the background while blurring out the rice grains. It **returns a background-only map**.
- **Image Subtraction:** The isolated background map is subtracted from the original image. This operation **returns a foreground-isolated image** where the wavy interference is neutralized, leaving only the rice grains on a uniform black field.

0.3.2 Final Segmentation and Results

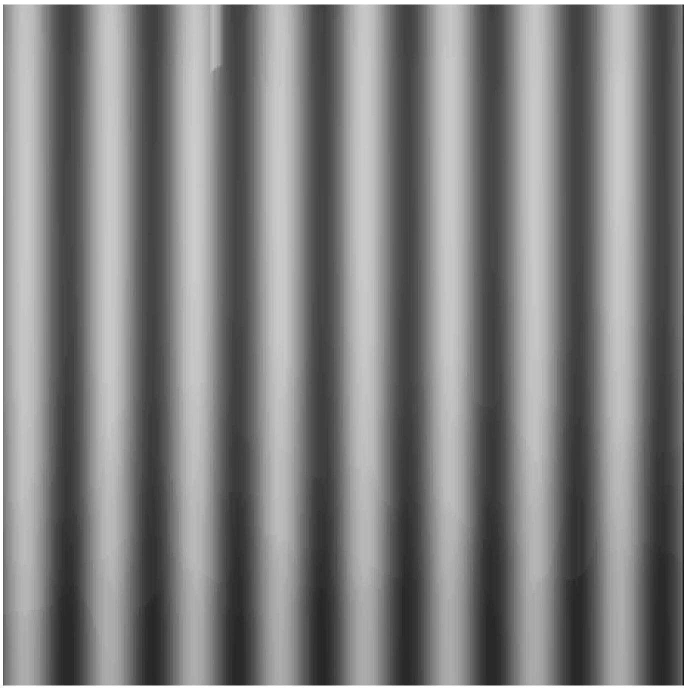
After the background is suppressed, the following steps are performed to finalize the count:

1. **Otsu's Thresholding:** Automatically determines the optimal threshold to convert the cleaned grayscale image into a binary mask.
2. **Morphological Opening:** Removes any remaining artifacts or small pixel noise from the subtraction process.
3. **Watershed Algorithm:** Separates individual grains that are physically touching to ensure an accurate tally.

Final Result: The implementation of this pipeline successfully isolates the grains, returning a final count of **97** grains for Pic 3.

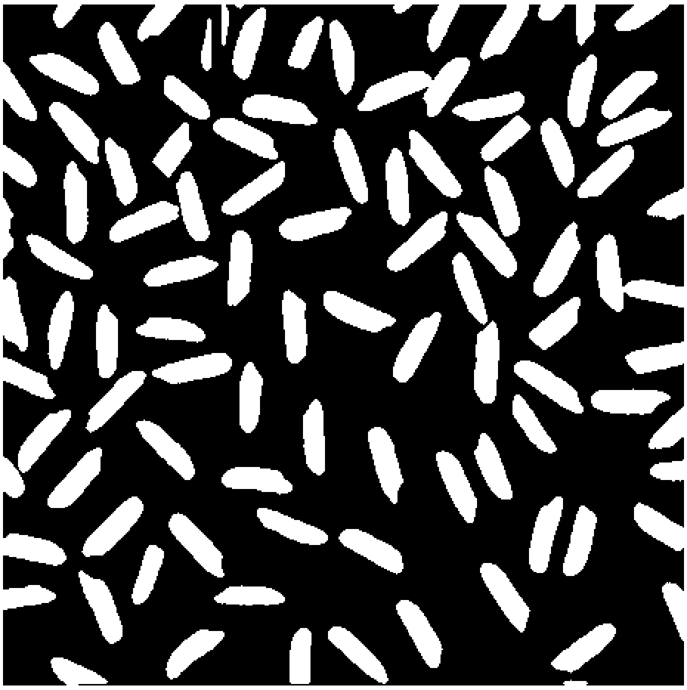


Hình 0.12: Image 3 Grayscale

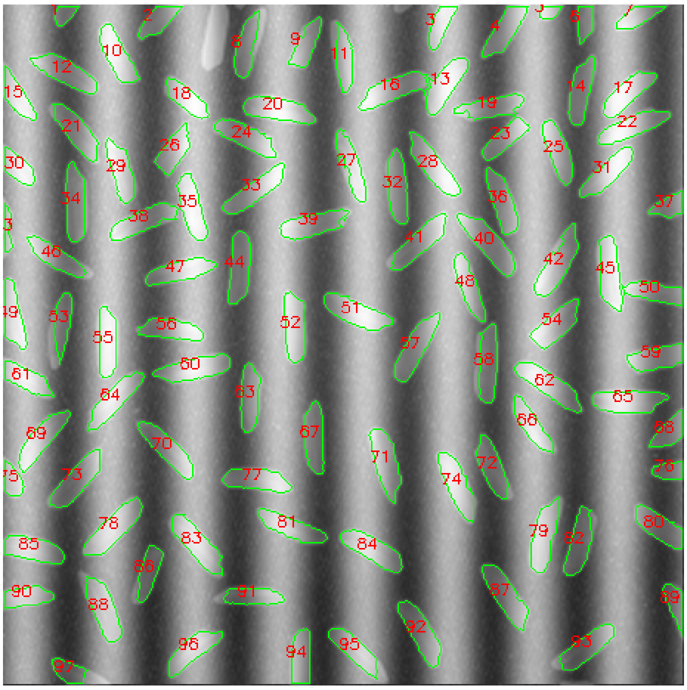


Hình 0.13: Image 3 Background Waves

Hình 0.14: Image 3 Normalized



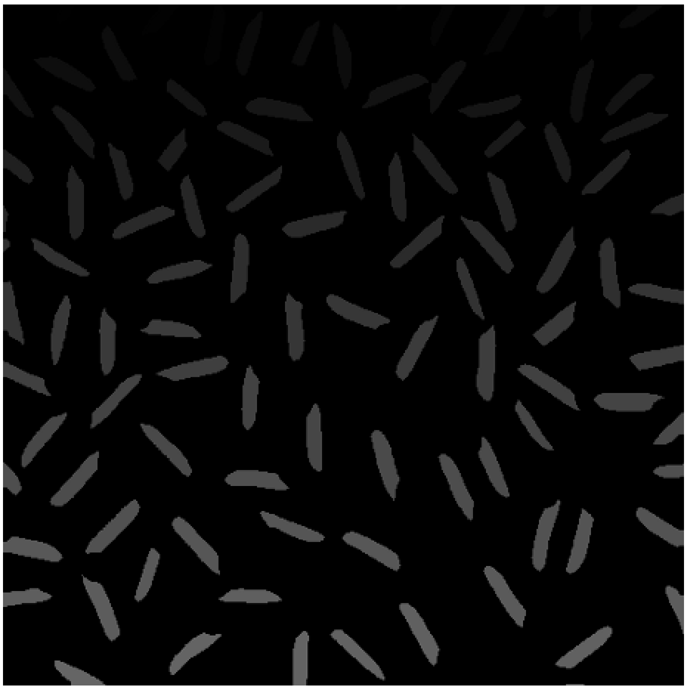
Hình 0.15: Image 3 Grayscale



Hình 0.16: Image 3 Results

0.4 Final Segmentation and Cluster Separation (Pic 4 & Universal)

The final stage of the counting system addresses the challenge of "low contrast" and "clustered objects." In Image 4, while the background is relatively clean, the grains are dim and often touch each other, requiring sensitive thresholding and robust separation logic.



Hình 0.17: Original Image 4 featuring low-contrast grains and clusters

0.4.1 Segmentation Logic

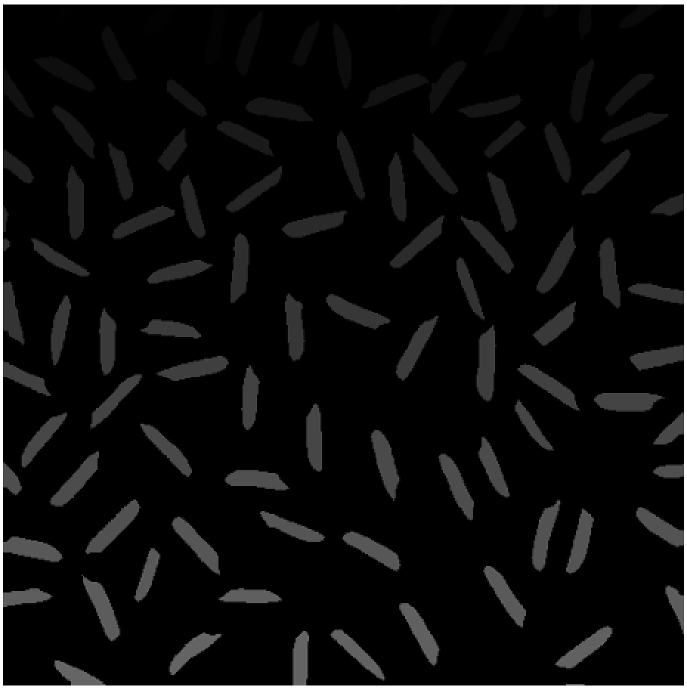
The following steps are critical for turning a grayscale image into a verifiable count:

- **Otsu's Thresholding (Key Step):** This algorithm analyzes the image histogram to find the optimal intensity value that separates the dark grey rice from the black background. It **returns a binary mask** representing the presence of rice.
- **Morphological Opening:** This operation performs erosion followed by dilation. It **returns a cleaned binary mask** by removing tiny stray pixels and smoothing the boundaries of the grains.

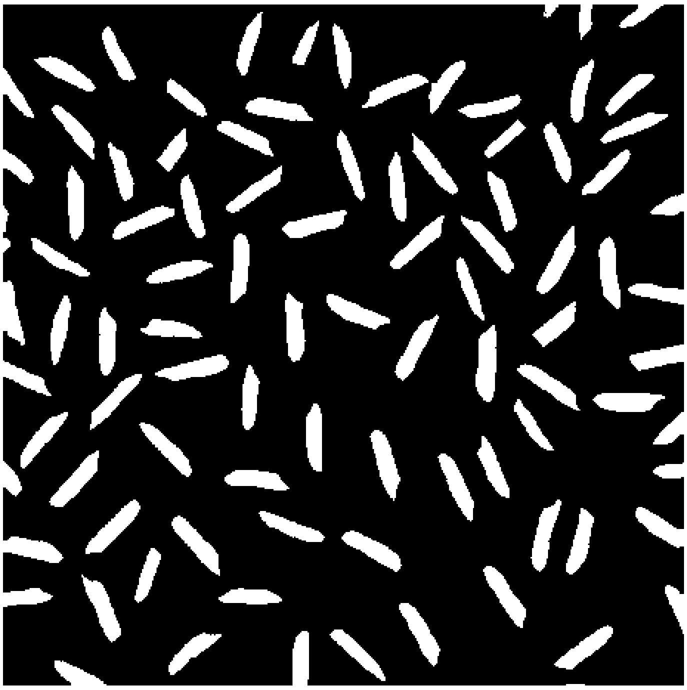
0.4.2 The Watershed Algorithm for Touching Grains

When grains are touching, they appear as a single large object in a binary mask. To fix this, the Watershed algorithm treats the distance transform of the binary image as a topographic map.

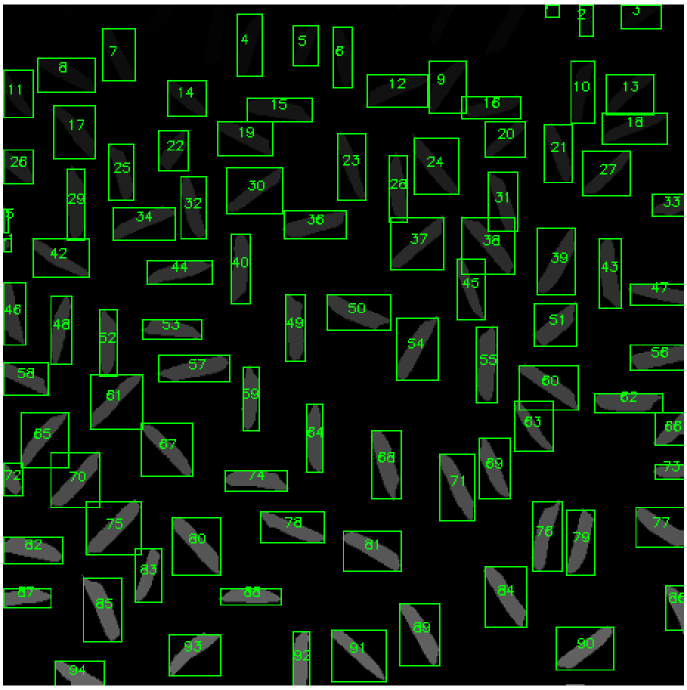
- **Distance Transform:** It calculates the distance from every white pixel to the nearest black pixel, effectively finding the "peak" or center of each grain.
- **Marker-Controlled Watershed:** The algorithm "floods" from these peaks until they meet at the boundaries where grains touch. It **returns unique integer labels** for every individual grain.



Hình 0.18: Image 4 Grayscale



Hình 0.19: Image 4 Threshold



Hình 0.20: Image 4 Results

0.4.3 Summary of Classical Computer Vision Results

The following table summarizes the performance of the proposed classical pipeline across all test images provided in Project 1.2:

Sample	Primary Technique Used	Final Count
Image 1	Top-Hat (40×40) + Median Blur	98
Image 2	Top-Hat (40×40) + Median Blur	98
Image 3	Vertical Kernel (1×80) + Subtraction	97
Image 4	Otsu Thresholding + Watershed	94

Bảng 1: Quantitative results for rice grain counting