Applying Machine Learning for Chili Pepper Phenotyping and Feature Extraction

Trong Tai Ha[1,2†], Thi Nga Pham[1,2†], Thanh Tuan Thai[1,2,3], Tien Dung Mai[1,2], Duc Thanh Ngo[1,2,*], and Yong Suk Chung[3,*]

[1]Faculty of Computer Science, University of Technology, Ho Chi Minh city 70000, Vietnam. emailTrong-Tai@uit.edu.vn(T.T.);emailNga@uit.edu.vn(T.N.);tuantt@uit.edu.vn(T.T.T.);emailthayDung@uit.edu

[2]Vietnam National University, Ho Chi Minh city 70000, Vietnam.

[3]Department of Plant Resources and Environment, Jeju National University, Jeju 63243, Republic of Korea.

[*]Address correspondence to: yschung@jejunu.ac.kr (Y.S.C.); thaythanh@uit.edu.vn(N.D.T)

[†]These authors contributed equally to this work.

Abstract

Traditional methods of phenotyping and accurately characterizing chili pepper fruit morphology include manual inspection and measurement by trained personnel, which is labor-intensive, time-consuming, and error-prone due to human error and subjectivity. With rapid advances in the fields of computer vision and machine learning, we propose a method that uses machine learning and computer vision to automate the process of phenotyping and extracting morphology features of chili peppers and establishing an information retrieval dataset for the chili pepper variety. We introduce a model for phenotyping and extracting features of chili peppers based on longitudinal slice images of chili peppers and introduce a set of image processing techniques. To solve the challenge of determining the location of chili peppers and seeds, we use the YOLOv7 [1] model. After retraining on the (Chili Pepper) Dataset, the model showed significant results in determining the location of chili seeds and chili pepper. After that, we separate each chili and use machine learning techniques to extract further information on characteristics that contribute significantly to evaluating the chili, such as counting the number of chili seeds, quantifying color, length, width, area of the chili fruit, and wrinkles of the fruit shape. Finally, we introduced the first model that can help identify phenotypes and extract characteristics of chili peppers automatically through longitudinal slice images of chili peppers. The first test results on the YOLOv7 model for accuracy and mAP are (0.92, 0.87). Then, the parameters are extracted based on the defined calculation functions. Parameter results are saved to support storing phenotypic information and characteristics of each chili variety, enabling efficient data management

and analysis for breeding programs.

1 Introduction

Chili pepper, cultivated worldwide and used for thousands of years [2], is a pungent fruit from the Solanaceae family. Chili peppers are valued for their distinctive flavors, nutritional properties, and medicinal benefits [3]. They are an excellent source of various vitamins, including vitamins E, C, A, and B complex, and minerals such as thiamine, folate, molybdenum, manganese, potassium, calcium, and iron. Additionally, chili peppers contain polyphenols (mainly luteolin), flavonoids, and quercetin. In many regions, chili peppers play a pivotal role in local cuisines, imparting unique tastes and adding depth to traditional dishes. Beyond culinary applications, chili peppers are also utilized in various industries, including pharmaceuticals, cosmetics, and even self-defense products, due to the presence of capsaicinoids – the compounds responsible for their characteristic pungency. The chili pepper market has experienced significant growth, driven by increasing consumer preferences for diverse and authentic flavors and the recognition of the potential health benefits associated with capsaicinoids. Their widespread use as condiments and functional food ingredients has increased global demand for fresh and processed chili pepper products, creating opportunities for growers, processors, and traders. Furthermore, chili peppers have become an important constituent in many industrial applications. Their popularity extends beyond culinary uses, as they are utilized in pharmaceuticals, cosmetics, and even self-defense products, owing to capsaicinoids.

The great diversity of Capsicum varieties poses challenges. With wide varieties of chili peppers available, each with unique distinct phenotypic characteristics in shape, size, color, spiciness, and flavor, it presents opportunities and challenges for varietal management and breeding programs [4]. Accurate and efficient characterization of these phenotypic features is pivotal for unlocking the full potential of chili pepper varieties and driving targeted breeding efforts. Traditional methods of phenotyping and accurately characterizing chili pepper fruit morphology include manual inspection and measurement by trained personnel, which is labor-intensive, time-consuming, and prone to human error and subjectivity. Moreover, these manual approaches often lack the precision and consistency required for comprehensive varietal analysis and comparison.

The digitization of phenotypic features [5–7] through advanced imaging techniques and computer-aided analysis offers a transformative solution to these challenges. Researchers can quantify and extract numerical features with unprecedented accuracy and objectivity by capturing high-resolution images of chili pepper fruits and leveraging machine learning algorithms. This digitized approach facilitates the precise measurement of traits such as fruit dimensions, seed count, color parameters, and other relevant morphological and biochemical characteristics. The resulting numerical data enables detailed varietal profiling and supports data-driven decision-making in breeding programs. Furthermore, the digitization of phenotypic features allows for the creation of comprehensive databases, enabling efficient storage, retrieval, and analysis of varietal information. This data-driven approach empowers breeders to identify desirable traits, assess genetic diversity, and make informed selections for developing new varieties tailored to specific market demands or environmental conditions. By embracing the digitization of phenotypic features, the chili pepper industry can unlock new avenues for varietal management, accelerate breeding cycles, and drive the development of improved varieties that meet the evolving needs of consumers and stakeholders alike.

During the research process, three main challenges have been identified in constructing the dataset and extracting feature information of chili peppers:

- **(a)** The lack of standardized and diverse datasets poses a significant challenge in chili pepper detection and phenotyping research using computer vision and machine learning techniques. Many existing datasets are created and annotated by individual research groups with varying protocols, leading to inconsistencies in data representation and labeling conventions.

- (**b**) Handling occlusion and overlapping of chili seeds. The use of cross-sectional images causes chili seeds to overlap, making it difficult for the model to identify the chili seed subjects.

- **(c)** Unclear subject boundaries have long been a focus in the field of image retrieval. This factor often makes subject identification in images blurry and challenging. Specifically, unclear subject boundaries can complicate object extraction and even lead to incorrect search results.

According to the reference, for the first time, we introduced a model to help determine the location of chili peppers and chili seeds and extract information about species-specific characteristics through longitudinal slice images of chili fruits. Improvements were made in the way YOLOv7 was utilized. Then, the approach using machine learning techniques in image processing was adopted and adjusted following the defined context of chili fruit. Our goal was to build an accurate chili detection framework to address the complexity of determining phenotypes and characteristics of chili peppers. This aimed to provide robust technical support for practical applications in the agricultural sector, particularly for later storing and retrieving information about each chili variety. Our research aims to facilitate the easy storage and retrieval of information about the characteristics that affect the quality of each chili variety, thereby contributing to breeding efforts and promoting the sustainable development of the agricultural industry. For the first time, we introduced a model that can help identify phenotypes and extract characteristics of chili peppers automatically through longitudinal slice images of chili peppers, as described in Figure *.

This study's primary objective is to harness machine learning's capabilities, particularly the YOLO (You Only Look Once) object detection model, to precisely identify phenotypes and extract morphological features of chili peppers. We aim to develop a reliable and automated system by utilizing this advanced technology. More specifically, the research employs the YOLO model to accurately identify and locate crucial elements of chili peppers, such as seeds and the fruit itself, in high-resolution imagery. This object detection capability forms the basis for subsequent extraction of morphological features.

Through the analysis of identified objects, the study aims to extract and quantify significant morphological features. The parameters extracted include, but are not limited to:

- **Bounding box size:** Accurate measurement of the length and width of the fruit detection frame box.

- **Fruit surface area:** Calculation of the fruit's surface area, which is relevant data for evaluating potential yield and considering processing.

- **Color parameters:** Quantifying the color characteristics of the fruit, particularly the degree of redness, as an important indicator of ripeness and quality.
- **Fruit size:** Precisely measuring the length and width of the fruit, enabling reliable comparison of morphological characteristics across varieties.
- **Number of seeds per chili:** Accurately determining the number of seeds per chili, a trait vital for yield and genetic diversity.
- **Edge wrinkles:** Measuring the wrinkled edges of the maturing fruit, a telltale sign of optimal ripeness and peak flavor, to assess the quality and consumption readiness.

By achieving these objectives, the research aims to provide a robust and scalable solution for extracting valuable numerical data from chili peppers, streamlining the varietal characterization process, and enabling data-driven decision-making in breeding programs and seed management strategies.

2 Materials and Methods

2.1 Dataset

The database we employed in this paper comprises cross-sectional images of chili peppers and meticulously labeled seeds. These chili peppers were not acquired from a commercial source. They were instead grown and collected at the Rural Development Administration (RDA), located in Jeonju, Republic of Korea, under our direct supervision. This allowed us to maintain a high level of control over the growth conditions and the overall quality of the chili peppers.

Cultivation and Sample Selection

The cultivation of the chili varieties was conducted under strictly controlled environmental conditions. We carefully monitored and regulated the chili plants' temperature, irrigation, and fertilization. This level of control was crucial in maintaining the uniformity of our samples and reducing any potential bias in our data caused by environmental variations. We experimented in a garden of 77 chili species, each bringing unique characteristics to our study. Each species was randomly arranged in plots within our greenhouse and assigned a unique species code (Figure *) for easy identification and data tracking.

Our selection process of the chili peppers was meticulous and based on standardized criteria. We considered factors such as the area and weight of the chili peppers, their color, and other relevant characteristics. This allowed us to maintain a high level of consistency in our sample selection and ensured the reliability of our data. We planted six trees per sample plot for each species and collected six fruit samples. The sampling process was initiated when the plants had fully fruited and reached maturity. This ensured we assessed the chili peppers at a standardized growth cycle stage.

Image Acquisition

We used The camera system settings based on the method described by Yu et al [8]. (2023). By adhering to these settings, we aimed to ensure consistency and reliability in our image data. We captured the images using a Canon digital camera. We emphasized maintaining uniform lighting conditions during the image capture and kept distances and shooting angles consistent. This was done to eliminate potential bias or variability in the image data from changing lighting conditions or different shooting angles. The images were saved in JPG format, preserving the intricate details of the chili peppers with a pixel resolution of 6024×4024. The database contains a total of 32 images before augmentation.

Annotation and construction

We used a cropping technique to extract 640x640 pixel squares from the original image, ensuring each cropped region contained at least one chili pepper. This preprocessing step simplified subsequent image-processing tasks and optimized the computational resource utilization during model training and inference. We leveraged the Roboflow tool for the labeling process. Roboflow is a powerful and user-friendly platform that streamlines the task of annotating objects within images, a crucial step in preparing data for training machine learning models, especially in computer vision.

We defined two classes for our labeling task: "chili" and "seed." The "chili" class represented the entire chili pepper fruit, while the "seed" class included the individual seeds within each chili pepper. Labeling involved manually drawing bounding boxes around each chili pepper and seed instance in the images. We employed a team of trained annotators who followed strict guidelines to ensure consistency and accuracy in the labeling process. For the "chili" class, the annotators were instructed to draw tight bounding boxes around the chili pepper fruit. For the "seed" class, they drew bounding boxes around each seed visible within the chili pepper. The Roboflow tool provided a user-friendly interface for annotating the images, with features such as zooming, panning, and keypoint labeling. It also allowed us to review and correct any inconsistencies or errors in the annotations before finalizing the dataset. Once the labeling process was complete, we exported the annotated dataset from Roboflow in the YOLOv5 format, which is compatible with most popular object detection frameworks and model architectures.

To further enrich the diversity of our dataset, we incorporated various augmentation techniques, including rotation and flipping images. These data augmentation strategies aim to synthetically expand the training data by introducing variations in orientation and other image characteristics. Exposing the model to a wider range of data during training enhances its ability to generalize and improve its performance on unseen samples.

The dataset includes more than 700 annotated images of different pepper varieties taken under the same distance, lighting conditions, and background. The images were manually labeled with bounding boxes around each chili sample and seed. We split our annotated dataset into train, validation, and test sets to ensure robust model training and performance evaluation. This partitioning is a common practice in machine learning to prevent overfitting and assess the generalization capability of the trained models. We followed a 70-20-10 split ratio, allocating 70% of the data for training, 20% for validation, and 10% for testing purposes. The training set was used to optimize the model's parameters during training. In contrast, the validation set was employed for hyperparameter tuning and monitoring the model's performance during training to prevent overfitting. The test set, kept entirely separate from the training process, was an unbiased measure of the model's performance on unseen data. By evaluating the model's predictions on the test set, we could obtain a realistic estimate of its generalization capabilities and assess its suitability for deployment in real-world scenarios.

The training, validation, and test sets were carefully curated and organized, with the image filenames and corresponding annotations stored in separate files for each subset(Figure). This structured organization facilitated efficient data loading and preprocessing during our machine learning pipeline's training and evaluation phases.

2.2 Model Training and Inference

We utilized the YOLOv7 [1] object detection model to accurately identify and locate chili peppers and their seeds. YOLOv7 is a version of the popular YOLO (You Only Look Once) [9] family of real-time object detection systems developed by Ultralytics. Its advanced model architecture achieves high accuracy while maintaining computational efficiency, making it well-suited for our application. The YOLOv7 model was pre-trained on the COCO dataset, a large-scale object detection, segmentation, and captioning dataset, comprising over 330,000 images across 80 object categories.

To adapt the model to our specific task of detecting and identifying chili pepper phenotypes, we employed transfer learning and fine-tuned the model. The pre-trained weights from the COCO dataset were used as initial values for the model parameters. We then trained the model on our chili dataset, which contained annotated images of various chili peppers with bounding boxes around the fruits and seeds. After this fine-tuning process, the YOLOv7 model demonstrated excellent performance in accurately detecting and locating chili peppers and their seeds in new images.

The predicted results from YOLOv7 are used as input for computational blocks and serve as the basis for extracting relevant object features. We have two computational blocks as follows:

Chili Pepper Mask Block

We separate the chili pixels from the background by extracting the red channel from the RGB color space and then setting a threshold equal to the maximum value of the histogram plus 16. To refine the segmentation, a closing morphological operation is applied to fill in any holes or small gaps within the regions representing the chili. Subsequently, the OpenCV contour finding algorithm detects the contours of the connected components within the mask. This contour detection step facilitates the removal of any remaining noise or unwanted small regions by selecting the max contour. The final result is a precisely defined mask outlining the area of the chili fruit.

Ratio Block

To calculate the ratio between pixels and millimeters, we rely on the size of the QR code present in the image. Each image in our dataset is captured with a series of QR codes (Image*).

We use a function to read the dimensions of the QR code, then use the actual size obtained to calculate the ratio between pixels and millimeters, and return the value for this block.

Numerical Feature

For the `chili` class, we can calculate and convert features to milimeters use `ratio` calculated from Ratio Block.:

- **Bounding box size:** Calculate the `width_bbox` (width) and `height_bbox` (height) values of the bounding box cover the chili.
- **Fruit surface area:** `area_fruit` computed by the number of pixels in chili mask.
- **Color parameters:** Convert an image to the HSV color space and filter it using a mask for chili. Then, adjust the color channel values so that the redder the color, the larger the value. The `redness` (red level of chili) computed by sum of values within mask of chili.
- **Fruit size:** Calculate the `length_fruit` (actual length) and the `width_fruit` (average width) of the chili mask.
- **Edge wrinkles:** Calculate the `wrinkle` level of a chili using its contour (the edge of the chili mask)

For the `seed` class, we can extract features like:

- **Number of seeds per chili:** The `number_seeds` (number of seeds in each chili pepper) is determined by analyzing the spatial relationship between the chili pepper and its bounding box. For each "chili" bounding box predicted by YOLOv7, we count the chili seeds by examining coordinates. If the coordinates of a chili seed are within the chili pepper, the count increases by 1. We iterate through all the bounding boxes in this manner to obtain the result.

3 Results

Our results demonstrate the object detection models achieved high mean Average Precision (mAP) scores, enabling accurate localization of chili peppers and their seeds within images. The image analysis methods precisely predict relevant morphological features, while the quantitative color models agreed closely with actual ground-truth measurements. The model exhibited promising performance, providing valuable information for variety classification and quality assessment of chili peppers. The shape analysis techniques effectively quantified key morphological traits, including fruit size, surface area, and edge wrinkles, enabling reliable varietal characterization.

3.1 YOLOv7

The prediction results in the images below are obtained using the YOLOv7 [1] and YOLOv5 [10] object detection models. These models can detect and localize objects within the images by drawing

bounding boxes around them. In this particular case, the models are trained to detect two classes: "chili" and "seed".
Insert Image: Prediction results from YOLOv7
Insert Image: Prediction results from YOLOv5
As you can observe, the images display bounding boxes around the detected chili peppers and seeds. The YOLOv7 and YOLOv5 models are compared in their performance in accurately detecting and localizing these objects.

Table 1: Compare results of YOLOv5 and YOLOv7

	Precision	Recall	mAP@50	mAP@50-95
YOLOv5	1	2	3	4
YOLOv7	2	3	4	5

3.2 Orient Block

To ensure that the images of the cropped chili peppers are all oriented in the same direction, we use the region of the chili stem (the center of the chili stem) to determine whether the chili pepper is horizontal or vertical. If the chili pepper is horizontal, we rotate the image by right angle. The process involves the following steps:

- Determine the `GreenMask`: covers the stem area of the chili, filtered using the green filter of the HSV color space.
- Calculate the *Center* of `GreenMask`: Find the center of the chili stem by calculating the average position of all the pixels belonging to the stem.

$$\begin{aligned} Center_x &= \frac{\sum_{i=1}^{s} x_i}{s} \\ Center_y &= \frac{\sum_{i=1}^{s} y_i}{s} \end{aligned} \tag{1}$$

  Where s is the number of pixels belonging to the stem, $Center(Center_x, Center_y)$ is the central coordinate of the `GreenMask` in the Oxy coordinate.

- Determine orientation of chili pepper: Compare the position of the *Center* to the central position of image to determine either the chili is horizontal or vertical. The stem will be located at the top of the chili, so the position of the Center will usually be far from the middle along the chili's length and closer to the middle of the image when considering the chili's width.

$$\begin{aligned} \Delta_x &= |W - 2 \times Center_x| \\ \Delta_y &= |H - 2 \times Center_y| \end{aligned} \tag{2}$$

Where W, H are width, height of image size respective. Δ_x, Δ_y are twice the distance from the Center to the middle axis along the width and height of the image, respectively.

- Rotate Image: If $\Delta_x > \Delta_y$, the chili is horizontal and rotate the image at a right angle so that the chili stem points upwards. Otherwise, it is vertical and do not anything.

3.3 Chili pepper mask Block

First, for each chili pepper image, we segment the chili pepper using a `threshold`. The `threshold` is selected to ensure that the classification of the chili from the background yields the best results for most chilies in the dataset. We tested using grayscale images and each color channel of the RGB color space, and it showed that the threshold on the red channel yielded the best classification results.

$$\begin{aligned} Threshold &= argmax(histogram) + \gamma \\ \texttt{ChiliMask} &= segm(R, Threshold) \end{aligned} \tag{3}$$

Where *histogram* represents the brightness distribution in the red channel of the image. `ChiliMask` is a binary image that classifies the chili and the background using a *Threshold*. R R is the image of the chili in the RGB color space, but only the red channel is taken.
However, a major drawback is that the interior of the chili pepper, including seeds and the core, has yellow and white colors, causing noise and making it difficult to capture the entire surface of the chili pepper. As illustrated in the example Figure 1.
To refine and address this drawback, we use the closing method from the cv2 library on the mask

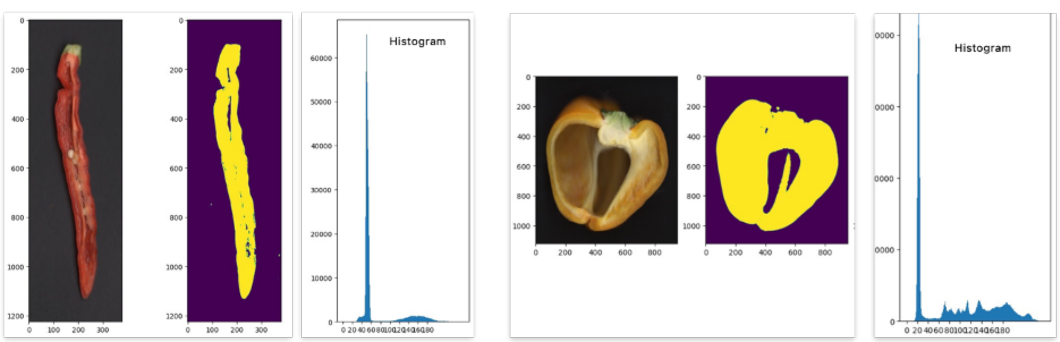

Figure 1: Illustration of Chili pepper mask with only *Threshold*

to fill in regions within the chili pepper that are not labeled as chili pepper. This helps to fill any holes or small gaps within the regions representing the chili pepper. (Figure 2)

$$\texttt{ChiliMask}_2 = CLOSE(\texttt{ChiliMask}) \tag{4}$$

Where $\texttt{ChiliMask}_2$ is $\texttt{ChiliMask}$ after apply $CLOSE$ function (closing method from the cv2 library)
Next, we use the find contours algorithm from cv2 library to retain the largest contour and remove

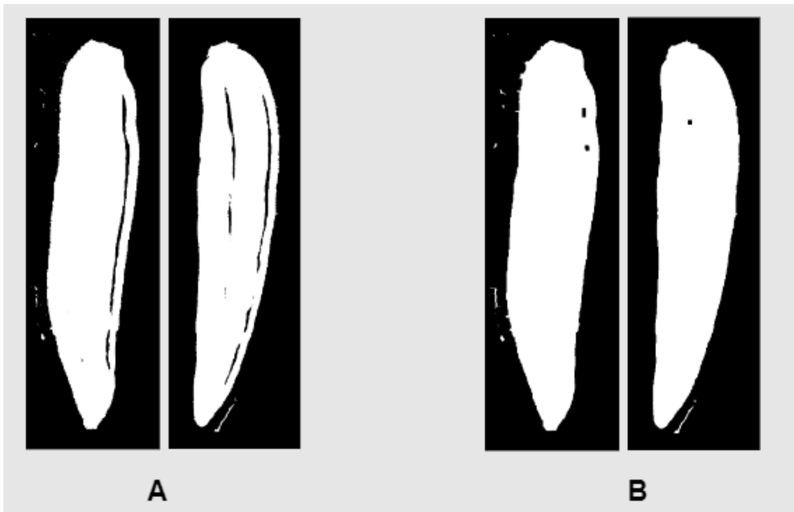

Figure 2: A is the chili mask obtained after applying thresholding for segmentation, and B is the result after applying the closing method to A, showing that the missed areas inside the chili have been filled.

the remaining contours. This helps to eliminate any pixels not within the contour region of the chili pepper but are labeled as chili pepper. The final output of this block achieves the desired segmentation, as shown in Figure 3.

$$\begin{aligned} Contours, hierachy &= FINDCONTOURS(\texttt{ChiliMask}_2) \\ \texttt{ChiliMask}_3 &= MAX(Contours) \end{aligned} \tag{5}$$

Where *Contours* is a list of points representing the boundary of an separated regions, *hierachy* provides the relationship between contours (such as parent-child relationships). The $FINDCONTOURS$ function is find contours method from cv2 library, MAX function get the *Contours* max. The $\texttt{ChiliMask}_3$ is filter the contour with the largest area from the $\texttt{ChiliMask}_2$.

3.4 Convert Pixel to Milimeter Block

To calculate the ratio between pixels and millimeters, we rely on the size of the QR code present in the images. Each image in our dataset is captured with a series of QR codes attached (Figure 4). We use a function to read the QR code and then determine the size of the QR code. Using the known real size (10 mm), we calculate the ratio between pixels and millimeters and return the value in mm/pixel for this block.

Given the QR code's size in pixels, $size_{QR}$, and the actual size of the QR code, 10 mm, the `ratio`

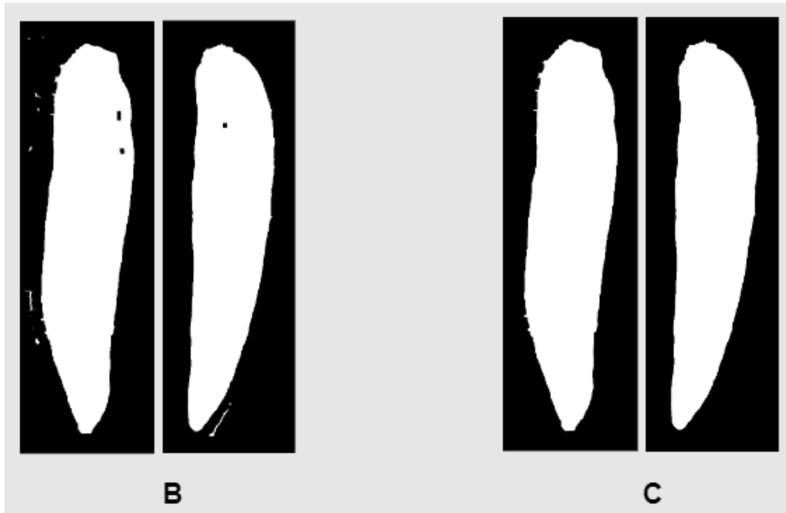

Figure 3: C is the result after applying the contour max method to B, showing that the noisy pixels outside have been removed.

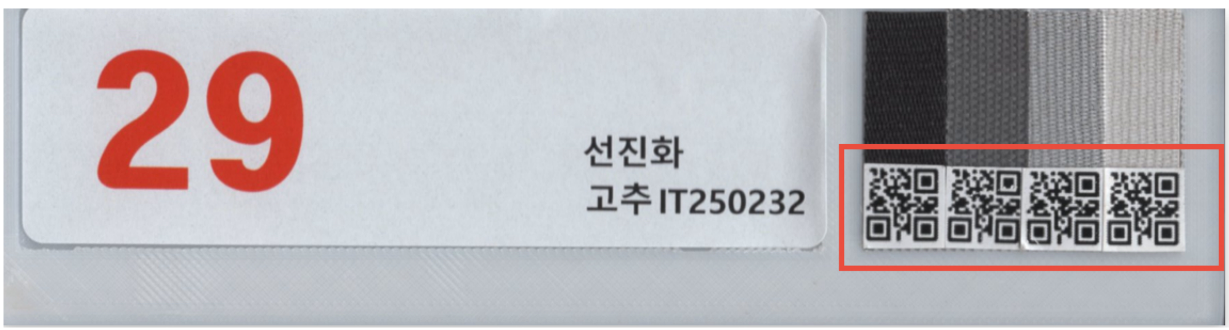

Figure 4: QR barcode

calculated and used convert measurements from pixels to millimeters in subsequent calculations.

$$
\begin{aligned}
size_{QR} &= READIMAGE(X) \\
\texttt{ratio} &= \frac{10\ (\text{mm})}{size_{QR}\ (\text{pixel})}
\end{aligned} \tag{6}
$$

Where X is the original image that has not been cropped into individual chilies and still contains the area with the QR code. The READIMAGE function read image X and return size of QR code follow unit pixel.

3.5 Numerical Feature Extraction

The bounding box predictions from the fine-tuned YOLOv7 model, Ratio Block, and Chili mask Block were the base for our numerical feature calculation pipeline. So, we computed the following numerical features:

Bounding box size

Through the output from YOLOv7 model, following formulas to calculate the width and height of the chili pepper's bounding box. Given a bbox (bounding box) represented by $[left, top, right, bottom]$, the width and height are calculated as follows:

$$\texttt{width\_bbox} = right - left \tag{7}$$
$$\texttt{height\_bbox} = bottom - top \tag{8}$$

Fruit surface area

To determine the area of the chili pepper, we count the number of pixels in the $\texttt{ChiliMask}_3$. Using the `ratio` to convert the surface area of the chili pepper from pixels to square millimeters. The area calculation can be performed as follows:

$$\texttt{area\_pixel} = \sum_{i,j} S(\texttt{ChiliMask}_3[i,j])$$
$$\texttt{area} = \texttt{area\_pixel x ratio} \tag{9}$$

Where S function return 1 if value input ≥ 1 and otherwise return 0

Color parameters

In the HSV color space, we use the hue channel to represent the color value and adjust this channel based on the redness, making the value larger as the color becomes redder. The `redness` equal average of hue's value in $\texttt{ChiliMask}_3$. Given a pixel value hue in the hue channel of the HSV color space:

$$\text{hue} = \begin{cases} 30 - \text{hue} & \text{if hue} < 30 \\ \text{hue} - 150 & \text{if hue} > 150 \\ 0 & \text{otherwise} \end{cases}$$

$$\texttt{redness} = \frac{\sum \text{hue}}{\texttt{area\_pixel}} \tag{10}$$

Fruit size

To calculate the `length_fruit` and `width_fruit` of the chili pepper, we combine the bounding box and the $\texttt{ChiliMask}_3$, which accurately reflects the discrepancy with the `height_bbox` and `width_bbox` parameter, thereby highlighting the curvature and deformation on both sides of the chili pepper. First, we divide the height of the bounding box into 9 equal parts using 10 horizontal dividing lines. The calculations are as follows:

$\texttt{width\_fruit}$: The average width of the $\texttt{ChiliMask}_3$ along the separation lines, excluding the topmost and bottommost separation lines.

$$\texttt{width\_fruit} = \frac{1}{8}\sum_{i=1}^{8}(\max\{j \mid \texttt{ChiliMask}_3[p, j] > 0\} - \min\{j \mid \texttt{ChiliMask}_3[p, j] > 0\})$$

$\texttt{length\_fruit}$: This is the length of the line connecting the midpoints of the segments containing the $\texttt{ChiliMask}_3$ and lying on the separation lines.

$$\texttt{length\_fruit} = \sum_{i=1}^{9}\sqrt{(\mathrm{p}[i-1][0] - \mathrm{p}[i][0])^2 + \left(\frac{\mathrm{p}[i-1][1] + \mathrm{p}[i-1][2]}{2} - \frac{\mathrm{p}[i][1] + \mathrm{p}[i][2]}{2}\right)^2}$$

Where $(p_{\mathrm{i}}, p_{\mathrm{i\text{-}1}})$ are the coordinates of the midpoints of the segments on each separation line that contain the chili pepper.

Number of seeds per chili

From the bounding boxes containing chili seeds, we calculate the centers of these bounding boxes. Given a bounding box for a seed represented by $\mathrm{bbox}_s = [left_s, top_s, right_s, bottom_s]$, the center $(x_{\mathrm{center}}, y_{\mathrm{center}})$ is calculated as follows:

$$x_{\mathrm{center}} = \frac{left_s + right_s}{2}$$
$$y_{\mathrm{center}} = \frac{top_s + bottom_s}{2}$$

Next, we iterate through these centers and count the number of seed centers that lie within the bounding box of the chili pepper. Given the chili pepper's bounding box $\mathrm{bbox}_c = [left_c, top_c, right_c, bottom_c]$, the condition to check if a seed center $(x_{\mathrm{center}}, y_{\mathrm{center}})$ is within the chili pepper bounding box is:

$$left_c \leq x_{\mathrm{center}} \leq right_c \quad \text{and} \quad top_c \leq y_{\mathrm{center}} \leq bottom_c$$

By iterating through all seed centers, the `nember_seeds` computed by count the number of seed centers that satisfy the above condition.

Edge wrinkles

To calculate the deformation of the chili pepper wall, since there has been no prior definition or formula for this measurement, we will present three methods that we have implemented and evaluated and provide reasons for selecting a single method for our study.

All three methods leverage the `contour` found from the $\texttt{ChiliMask}_3$.

1. **By Angles Formed by Three Consecutive Vertices**
   Analyzing the shape of the chili pepper based on contours is an essential method in computer vision for image recognition and classification. A practical approach to shape analysis is to

calculate the sum of the angles formed by three consecutive vertices on the chili pepper's contour and divide it by the number of vertices. This method is based on the observation that if the contour has many sharp bends, the angles formed by three consecutive vertices will be larger.

- **Calculate Angles Between Three Consecutive Vertices**:
  To calculate the angles formed by three consecutive vertices on the contour, we can use geometric formulas to find the angle between two vectors. - Suppose we have three consecutive points A, B, C on the contour. The vectors from A to B and from B to C can be calculated. Then, the angle between these two vectors can be calculated using the dot product formula:

  $$\theta = \cos^{-1}\left(\frac{\vec{AB} \cdot \vec{BC}}{|\vec{AB}||\vec{BC}|}\right)$$

  where

  $$\vec{AB} = (B_x - A_x, B_y - A_y)$$

  $$\vec{BC} = (C_x - B_x, C_y - B_y)$$

  and

  $$\vec{AB} \cdot \vec{BC} = (B_x - A_x)(C_x - B_x) + (B_y - A_y)(C_y - B_y)$$

  $$|\vec{AB}| = \sqrt{(B_x - A_x)^2 + (B_y - A_y)^2}$$

  $$|\vec{BC}| = \sqrt{(C_x - B_x)^2 + (C_y - B_y)^2}$$

- **Sum of Angles and Average Calculation**:
  Sum all the angles formed by three consecutive vertices on the contour.Then, the `wrinkle` computeed by devide the sum of these angles to the number of vertices in the contour to find the average value of the angles.

  $$\texttt{wrinkle} = \frac{1}{N}\sum_{i=1}^{N}\theta_i$$

This method provides a measure of the deformation of the chili pepper wall based on the sharpness of the bends in its contour.

2. **By Smoothness of Contour**
   In analyzing the shape of the chili pepper, we can use image processing techniques to find the contour of the chili pepper and calculate its characteristic parameters. An effective approach is to use the approxPolyDP function in OpenCV to approximate the contour and then calculate the smoothness and wrinkle measure based on this contour.

   - **Contour Approximation**:
     Use the approxPolyDP function to approximate the contour of the chili pepper. The

epsilon parameter in approxPolyDP determines the accuracy of the approximation and needs to be manually selected.

$$\text{approxContour} = \text{approxPolyDP}(\texttt{contour}, \epsilon)$$

Where ϵ is level approximate for `contour`.

- **Calculate Smoothness and Wrinkle Measure**:
  Smoothness is calculated as the ratio of the length of the approximated contour to the length of the original contour.

$$\text{Smoothness} = \frac{\text{Length of approximated } \texttt{contour}}{\text{Length of original } \texttt{contour}}$$

$$\texttt{wrinkle} = 1 - \text{Smoothness}$$

This method provides an effective measure of the smoothness and wrinkles of the chili pepper's shape based on its contour.

3. **Using Contour Over a Defined Segment**
In this method, we focus on calculating the number of bends in the contour over a defined length segment. We will calculate the number of bends in the `contour` of fixed length and then average them to get the final wrinkle measure.

- **Define the Segment**:
  The defined segment is determined by using the total length of the contour and dividing it into segments of fixed length. Calculations are based on these fixed segments.
- **Calculate the Number of Bends in the Defined Segments**:
  By calculating the angles as in Method 1, we can identify a bend if the calculated angle is greater than 130 degrees.

$$\theta = \cos^{-1}\left(\frac{\vec{AB} \cdot \vec{BC}}{|\vec{AB}||\vec{BC}|}\right)$$

  Where a bend is identified if $\theta > 130^\circ$.
- **Calculate the Total Number of Bends and Average**:
  Sum the number of bends over all segments and then average them to obtain the wrinkle measure of the contour.

$$\texttt{wrinkle} = \frac{1}{N}\sum_{i=1}^{N} \text{Number of bends in segment}_i$$

This method provides an effective measure of the contour's wrinkliness by focusing on defined length segments and calculating the number of significant bends.

Conclusion

Calculating the number of bends in the contour over defined length segments offers a practical method for analyzing the wrinkles of the chili pepper's edge. By focusing on segments and identifying significant bends, we can obtain a detailed measure of the contour's irregularities. This approach allows for precise classification and recognition of different chili pepper shapes based on their contour characteristics.

3.6 Analysis and Interpretation

4 Conclusion

The research introduces a groundbreaking model for phenotyping and characterizing chili pepper varieties using machine learning and image analysis techniques. The model utilizes the YOLOv7 object detection algorithm, retrained on a specialized chili pepper dataset, to accurately detect and locate chili peppers and seeds in cross-sectional images, automating a process traditionally reliant on manual inspection. By integrating object detection with advanced image processing techniques, the model can extract a comprehensive range of phenotypic traits, including length, width, area, color intensity, seed count, and peel wrinkles. The extracted data is efficiently managed and stored in a structured CSV format, facilitating easy retrieval and integration with existing databases and breeding programs. This capability to describe and compare chili pepper varieties based on their phenotypic traits aids in variety validation supports breeding efforts and contributes to the preservation of genetic diversity. The research findings have significant implications for germplasm management, targeted breeding, quality control, climate adaptation, and intellectual property protection in agriculture. While the model marks a significant advancement, the thesis also acknowledges its limitations and suggests future research directions, such as expanding dataset diversity, integrating multimodal imaging techniques, incorporating genetic and environmental data, and improving scalability and computational efficiency. Overall, this model revolutionizes the phenotyping and characterization process, offering accurate and comprehensive phenotypic profiles that can drive innovation and promote sustainable practices in the chili pepper industry.

Acknowlegments

References

1. Wang CY et al. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. IEEE 2022.
2. Idrees S, Hanif MA, Ayub MA, Hanif A, and Ansari TM. Chili Pepper. In: *Medicinal Plants of South Asia*. Ed. by Hanif MA, Nawaz H, Khan MM, and Byrne HJ. Elsevier, 2020:113–24.
3. Benkeblia Z. Medicinal uses and health benefits of chili pepper (Capsicum spp.): a review. MOJ Food Processing Technology 2019.
4. Waongo L, Kiebre M, Sawadogo B, Ouedraogo MH, Kiebre Z, and Bationo-Kando P. Phenotyping of local chili pepper (Capsicum spp) accessions cultivated in Burkina Faso. International Journal of Agricultural Policy and Research 2021;9:16–23.
5. Nankar AN, Tringovska I, Grozeva S, Todorova V, and Kostova D. Application of high-throughput phenotyping tool Tomato Analyzer to characterize Balkan Capsicum fruit diversity. Scientia Horticulturae 2020;260:108862.
6. R.Rodríguez G et al. Tomato Analyzer: A Tool for the Semiautomated Morphometric Analysis of Fruit Shape. Journal of Visualized Experiments 2010.
7. Khokhar ES et al. High-throughput Characterization of Fruit Phenotypic Diversity among New Mexican Chile Pepper (Capsicum spp.) Using the Tomato Analyzer Software. HortScience 2022.
8. Yu JK, Chang S, Han GD, et al. Implication of high variance in germplasm characteristics. Scientific Reports 2023;13:515.
9. Redmon J, Divvala S, Girshick R, and Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. URL: `https://arxiv.org/abs/1506.02640`.
10. Jun-Hwa Kim Namho Kim YWP and Won CS. Object Detection and Classification Based on YOLO-V5 with Improved Maritime Dataset. Journal of Marine Science and Engineering 2022.