International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

A Novel Method for Accurate Tree Enumeration in Development Projects Using Canny Gaussian Hysteresis Contour (CGHC)

Ashok Reddy Kandula[1], Raghu Datta Muthevi[*,2], Venkata Reddy Tiyyagura[3], Revanth Tatineni[4], Aravind Madhu Sudhan Padamata[5], Chandra Shekhar Kori[6]

Submitted:14/03/2024 **Revised**: 29/04/2024 **Accepted**: 06/05/2024

Abstract: We provide a new method for counting trees in development projects by utilising cutting-edge image analytics tools to provide accurate and useful information. We use satellite images or aerial photos, convert them to grayscale, and then use Gaussian blurring to minimise noise and improve clarity. By utilising Canny edge detection, we are able to discern tree outlines with remarkable precision, even registering minute differences in plant density. Our method's core is the careful computation of tree area from the contours that were recovered; this yields a numerical representation of the amount of trees in the region that was studied. We use a color-coded visualisation method to improve interpretability, designating areas with high tree density in vivid green, areas with moderate coverage in yellow, and places with sparse distribution in red. Our approach goes beyond simple counting by superimposing tree outlines over the original photos, which helps with spatial analysis and decision-making. Stakeholders may visually evaluate the distribution of trees, spot vegetation gaps or clusters, and make well-informed decisions for land diversion while taking conservation and environmental effect into account. To summarise, our novel methodology provides a thorough approach to exact tree counting in development projects, equipping stakeholders with correct information for sustainable land use and environmental management. Here we depict our project implementation: github.com

Keywords: *Computer Vision, Tree Enumeration, Canny Algorithm, Gaussian filter, Hysteresis, Contour Algorithm*

Introduction

In the complex realm of land development, where progress and environmental conservation intersect, accurate tree enumeration assumes paramount importance. When forested land must be diverted for infrastructure projects or urban expansion, understanding the existing tree population becomes critical. But manual surveys that are done the old-fashioned way take a lot of time, labour, and are prone to mistakes. We offer a novel approach that makes use of image analytics to get beyond these restrictions. By leveraging satellite imagery or aerial photographs, our system automates the process of counting trees. The primary objectives are twofold: first, to detect and identify trees within designated forest areas accurately, and second, to provide detailed information about the tree population. Our method entails creating a strong computer vision algorithm capable of analysing a wide range of images. This algorithm must take into consideration changes in tree species, sizes, and environmental circumstances. We gain significant insights from tree photos processed with the OpenCV package. The system counts the number of trees in certain locations and classifies them according to key factors such as species or diameter (girth) [1]. Validation techniques assure the precision of our image analytics technology. We evaluate its dependability by comparing the results to ground truth data gathered through manual surveys. The end result is precise data that provides actionable information for stakeholders such as environmentalists, lawmakers, and project managers [6].

An image analytics solution is being proposed herein which could make tree enumeration more efficient and accurate in forest land diversion projects. It focuses on automating the process of identifying trees from image data, validating the accuracy of that, and providing detailed information about the tree populations therein. To be efficient, scalable and seamlessly integrated with existing systems, this solution has visualizations for easy understanding upon use. In terms of ethics it adheres to privacy regulations and respects nature as much as possible. For informed choice about land use, environmental impact assessment while conserving them among others, this solution would be indispensable in promoting responsible development that can be sustained over time.

Forested lands are pivotal in the delicate dance between progress and environmental stewardship. The volume of trees growing on a piece of land is important when development projects necessitate diversion of forest

[1] *Seshadri Rao Gudlavalleru Engineering College, 521356, India.*
ORCID ID : 0000-0001-6741-8602
[2] *Seshadri Rao Gudlavalleru Engineering College, 521356, India.*
ORCID ID : 0009-0003-8040-4960
[3] *Seshadri Rao Gudlavalleru Engineering College, 521356, India.*
ORCID ID : 0009-0000-1310-5651
[4] *Seshadri Rao Gudlavalleru Engineering College, 521356, India.*
ORCID ID : 0009-0006-4656-4356
[5] *Seshadri Rao Gudlavalleru Engineering College, 521356, India.*
ORCID ID : 0009-0003-8095-2190
[6] *Seshadri Rao Gudlavalleru Engineering College, 521356, India.*
ORCID ID : 0009-0009-5629-3926
* *Corresponding Author Email: dattamuthevi@gmail.com*

lands. Traditional manual surveys take too long and contain mistakes. This challenge motivates us to develop this proposal. We suggest image analytics as an innovative solution. We can count trees automatically by using satellite imagery or aerial photographs. Efficiency and precision are what we are striving for. Our computer vision algorithms make quick work of vast landscapes, identifying trees with perfect accuracy. Therefore, their consequences help us make informed choices about land usage, conservation initiatives, and environmental impact assessments. It also aligns our project with other development needs while trying not to disrupt the ecological balance in order to promote sustainable land use practices for future generations that will live here or visit it in years to come.

The abecedarian thing behind this trouble is to address the issue of unethical deforestation in timbers. We want to regulate deforestation in an ethical way, similar as limiting the quantum of trees diced down. By resolving these issues, the proposed image analytics approach would allow for more effective and accurate tree recitation for timber land diversion. It'll offer stakeholders with critical information for making educated opinions about land use, environmental impact assessments, and conservation enterprise, guaranteeing responsible and sustainable development. Our solution automates tree counting using satellite imagery or aerial photographs. It replaces the labour- intensive manual surveys, saving time & resources. We develop a robust algorithm for analyzing diverse imagery. The algorithm accounts for variations in tree species, sizes, and environmental conditions. By processing contour data, we compute the total tree area within specified regions. This information provides a quantitative understanding of the tree population. We assign color codes based on percentage tree coverage:

- Green (≥ 70%)
- Yellow (30% to 70%)
- Red (< 30%).

Overlaying these color-coded contours on original images visually represents tree distribution. Our solution equips stakeholders with accurate data for informed decisions. It facilitates responsible land use and conservation efforts.

Our project bridges the gap between progress and ecological stewardship. It optimizes resource allocation while minimizing environmental impact. The following section of the study will include a literature review, identifying shortcomings in tree counting methodologies, and proposing for a new strategy. demonstrating their efficacy in the results section. A performance assessment compares accuracy, speed, and scalability with traditional methods. The conclusion evaluates the strategy's effectiveness and future potential, while the bibliography highlights the balance between ecological protection and development needs. This strategy tries to bridge the gap between development requirements and ecological protection.

1. Related works

Muhammad et al. [4] researchers solved the difficulty of counting olive trees over large areas. They recommended an automated system that uses sequential image processing techniques applied to a single red channel that is extracted from acquired images. The sealed boundaries which delineate tree perimeters are converted into white blobs via morphological reconstruction following edge detection and sharpening. When compared to ground truth data, circular blobs that meet specific criteria are classified as olive trees and have an estimate error of 1.27%.

Faezah et al. [5] study sought to measure tree species diversity at various heights in the forest. Two 20m x 20m plots were randomly split into four 10m x 10m portions, positioned at two elevations: the Suboh Trail (44.2m) and the Jelutong Trail (225m). At the lower level, 44 individuals from 15 species were gathered, whereas 41 individuals from 16 species were taken at the upper elevation. Gymnacranthera bancana was the most common species seen. Despite the fact that both routes had modest species richness, species representation was quite equal. These findings highlight the necessity of environmentally sound management and sustainable ecotourism in the Bukit Nanas Forest Reserve.

Chinsu et al. [6] using aerial colour photos, the empiricists attempted to improve the accuracy of tree crown demarcation in forest stands. To integrate picture texture information, the researchers developed the multi-level morphological active contour (MMAC) method. First, the RGB aerial bitmap picture was transformed to CIELAB, and prospective tree seeds were identified using simply the brightness value. The erroneous choices were then sorted out using vector quantization based on local texture. Finally, the MMAC algorithm was used to distinguish between distinct tree crowns. The results showed a 94% tree crown identification rate, with a 6% omission rate. Aerial colour photos with spatial resolutions of less than 0.5 metres showed potential for detecting complicated tree tops in hilly forest stands.

Chao et al. [7] case study monitors used deep learning models (ResNet-18, DenseNet-40) to classify tree species (maple), outperforming traditional methods. Segmenting Individual Tree Structures Automatically: PointNet++ used LiDAR point cloud data to successfully separate tree canopies, trunks, and branches. 4096 representative points resulted in optimal performance. Airborne Multi-Spectral LiDAR Point Clouds for Building Extraction: Using a deep learning architecture that incorporated CNNs and multispectral LiDAR data, buildings were reliably recovered. IoU (89.5%), F-measure (94.4%), completeness (93.7%), and accuracy (95.1An Autonomous Method for Locating and Quantifying Olive Trees: The estimate inaccuracy of an automated system that counted olive trees using remote sensing was 1.27% when compared to ground reality. Using Airborne Hyperspectral Images to Classify Tree Species: Across many datasets with different tree species, a spatial-spectral mixed network achieved high accuracy (93.31% to 98.82%). Utilising WorldView-2 Images to Classify Predominant tree varieties in a City Forest Park: The dominating species of trees in an urban forest park were classified with 80.81% accuracy.

Hui et al. [8] observers employed deep learning models such as ResNet-18 and DenseNet-40 will classify tree species (maple, pine, locust, and spruce) using high-resolution satellite imagery and aerial LiDAR data. The models' accuracies ranged from 86.9% to 90%. Notably, ResNet-18 performed better than conventional techniques. Accuracy was influenced by LiDAR intensity data and input picture size, highlighting the necessity of correct tree species categorization for conservation and planning.

Dong et al. [9] Deep Learning Approach on Cloud Data from LiDAR Points" examiners suggested a unique method based on the PointNet++ prototype. This method effectively splits the foliage, main stem, and limbs of trees using 3D laser scanning data. Data from 435 tree specimens, including species of Japanese larch, Red pine from Korea, and Pine from Korea, were used to empirically evaluate the prototype. with an ideal performance of 4096 representative points. Accurate tree structure segmentation is vital for forest management and lowering carbon emissions.

Dilong et al. [10] The inquisitors presented a unique deep-learning framework. To accomplish precise building segmentation, this technique integrates multispectral LiDAR data with convolutional neural networks (CNNs). The approach performed admirably, with 95.1% accuracy, 93.7% completeness, 94.4% F-measure, and 89.5% Intersection over Union (IoU) on two test regions. Accurate building extraction is critical for photogrammetry, remote sensing, and urban planning.

Jian et al. [11] investigators used numerous seasons WorldView-3 pictures from 2015, covering spring end, summer solstice, autumnal equinox and late autumn settings. Individual tree-based species categorization had the best overall accuracy when using summer solstice images, followed by spring end and autumnal equinox. Interestingly, classification accuracy was significantly improved when all three of these seasonal pictures were used together. The late-fall film was included, but it had no further impact on accuracy. These results emphasise that effective species identification in complex forest ecosystems requires the consideration of seasonal changes and the use of multitemporal data.

Table 1. Summary of Literature Survey

Author	Algorithm	Application	Key Findings	Accurac	Precision
Muhammad et al [4]	Arbor Crown Enumerator (ACE)	Olive tree detection and Enumeration	Olive Detection-Precision	96%	High
Faezah et al. [5]	Relative Dominance	Forest ecosystem diversity assessment	Bukit Nanas Biodiversity Tree Species	78%	Moderate
Chinsu et al. [6]	Nested Message Authentication Code (NMAC)	Individual tree delineation in forest stand	Aerial Tree Crown	94%	High

Chao et al. [7]	Classification and Regression Tree (CART)	Urban forest species classification	Tree Identification Comparison	87.1%	High
Hui et al. [8]	Convolutional Neural Networks (CNNs)	Individual tree species classification	Tree Recognition with LiDAR	90%	High
Dong et al. [9]	CPS Algorithm	Individual tree structure segmentation	LiDAR Canopy Segmentation	97%	95%
Dilong et al. [10]	SVM & CNN	Building LiDAR Extraction	Deep Learning Building Extraction	95.1%	High
Jian et al. [11]	Spectral Angle Segmentation & Supervised Scale Selection	Individual tree species classification	Enhancing Species Classification with Seasonal Photos.	72%	Moderate

2. Proposed Methodology

A. Existing Model:

The current model on which we worked contains the aerial bitmap images caused a textural shift in the MMAC algorithm, allowing for precise individual tree difference within forest communities. Forest inventory data is critical for effective forestry management, and remote sensing methods are crucial in collecting this data. The suggested approach improves the accuracy of tree delineation by using image texture features from aerial colour photos. By expanding the MMAC algorithm, it achieves better results for recognising individual tree tops, resulting in more precise estimations of forest volume, biomass, and carbon pool. The work bridges the gap between aerial and individual-tree-based approaches, enhancing forest density assessment and overall management [6].

B. Data Collection:

In the quiet theatre of data collection, our ballet unfolds—a delicate pas de deux between technology and nature. Our prima ballerina, the internet, surges with emerald waves, each pixel a silent ode to chlorophyll. We pirouette through official websites, capturing forest images—mosaics of leaves and light. But our dance extends beyond earthly bounds; we ascend on drone wings, viewing the forest from celestial heights—the green pulse of life beneath us. These snapshots, like ancient spells, reveal the contours of trees—their whispered secrets

embrace. In this choreography of science and wonder, we dance with pixels and leaves, celebrating the delicate balance between human curiosity and the forest's eternal mystery [2].

Fig.1. Data Collection Illustration

C. Data Preprocessing:

In order to create a captivating visual story, we developed a thorough data preparation method for our forest photographs. This method includes four main techniques:

- Selective Curation: We carefully selected photos that are consistent with our vision, ensuring that each one contributes to our narrative.
- Focus on Clarity: We prioritised clarity, removing any blurry photographs and concentrated on crisp, clear ones.
- Colour Normalisation: We used colour normalisation to maintain uniformity throughout our dataset. This

approach brings the colours of an image to a normal scale, improving overall visual harmony.

- Image Resizing: To ensure consistency and save computational effort, all photos have been reduced to standard dimensions.

This complete approach guarantees that our image data is not just high-quality but also purposeful, helping to create a unified and compelling visual narrative.

(a)
(b)

Fig.2(a). Collected Picture, 2(b) Preprocessed Picture

D. Data Visualization:

In the vast expanse of data, we start on a visual trip, travelling through pixel flora and fascinating patterns. Our dataset, analogous to a diversified forest, has 900 data points, each representing a distinct arboreal creature. These points, like leaves rustling in the breeze, are divided into three categories: training, cross-validation, and testing. Our tools are A palette inspired by computer science, statistics, and visual design. Univariate analysis focuses on specific traits including bark texture, leaf arrangement, and branch curvature. As we move through this dataset, we see patterns: gnarled roots of connection and sun-kissed leaves of significance. Beyond aesthetics, we are looking for resonance—a symphony of understanding. Our visualisations, like dew-kissed spiderwebs, convey complexity in digestible ways. Display formats such as scatter plots, dendrograms, and heatmaps complement the melody of the data. Interactivity entices people to investigate. Our canvas is more than just pixels; it is Processing, a fertile ground where code flowers into art. So let us roam, turning data into a forest of insights waiting to be uncovered.

Table 2. Data Points Visualization

`S. No	Data Set	Data Points
1	Training	2000
2	Testing	300
3	Validation	200
4	Total Data Points	2500

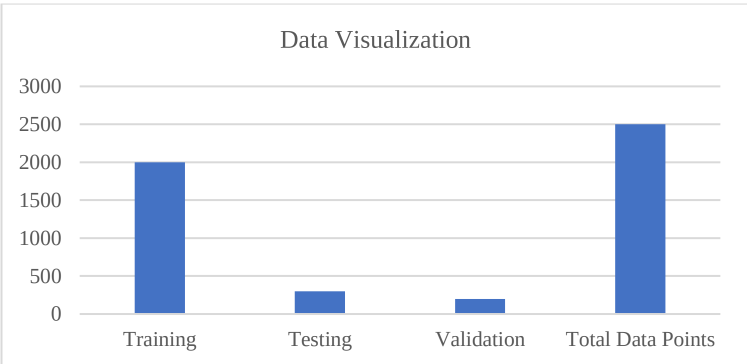


Fig.3. Bar Chart Visualization

E. Proposed Model:

The proposed model employs popular algorithms like Canny for edge detection and contour detection for tree morphology analysis. It preprocesses the image, removes the backdrop, identifies tree boundaries, and calculates the tree cover percentage by comparing contour and image areas.

F. Model Architecture:

This model design involves several stages. Initially, the image undergoes preprocessing to prepare it for analysis. Following this, the background is removed to concentrate on the objects of interest, in this case, the trees. The model then employs an edge detection technique, such as the Canny method, to identify the boundaries of the objects. These detected edges are analyzed to discern Contours that represent the shapes of the trees. In the final step, the model calculates the Percentage of the image occupied by trees by comparing the total area of the contours to the overall area of the image. The outcome of this process is a processed image and the calculated tree coverage percentage. Fig 4 embarks whole design of the prototype.

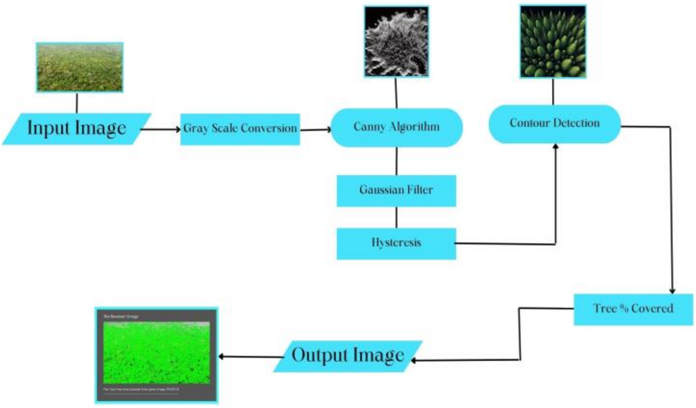


Fig.4. Design Diagram of the Project Proposed Methodology

3. Algorithms

A. Edge Detection Process:

Canny edge Discovery is a technology that excerpts structural information from visual objects, significantly reducing the volume of data to reuse. It has set up wide use in numerous systems based on computer vision. Canny discovered that the prerequisites for applying edge discovery to different vision systems are enough analogous. therefore, an edge discovery system to meet these conditions can be used in a variety of surrounds. The identification of edges with a low error rate is a general requirement for edge discovery, meaning that as many edges as possible should be reliably detected in the picture. The Canny edge detection technique consists of multiple phases. First, the supplied picture is transformed into grayscale. Next, a Gaussian blur is used to Minimise Noise. The intensity gradients are then determined using Sobel filters. Then, Non-Maximum Suppression is used to find local maxima in the Gradient Direction. Double thresholding aids in distinguishing between pixels with sharp edges and those without. Finally, edge tracking by Hysteresis guarantees continuous edge curves. Remember that adequate parameter tweaking is essential for obtaining accurate results. Figure 5(a) depicts the whole mechanism. It is renowned for producing thin, well-connected edges and for its accuracy in detecting edges with low error rates. It is extensively utilized in many different applications, including computer vision systems for feature extraction, image segmentation, and object detection. Canny works especially well in situations where precise edge detection with low noise is needed. Strong edge detection is ensured by its multi-stage process, even in the presence of changing noise levels and lighting conditions.

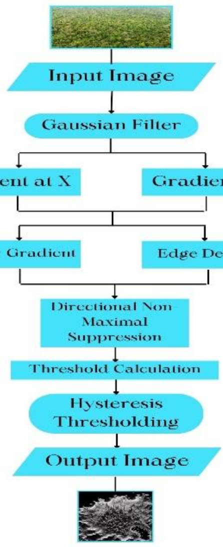


Fig. 5(a). Edge Detection (Canny) Process

B. Contour Detection Process:

The Contours Unveiled Amidst the pixelated tapestry of grayscale, where hues surrender to monochrome, contours emerge—an ethereal choreography etched upon the canvas of light. Each point, a sentinel guarding the boundary of existence, tracing the elusive edges of reality. Behold the CV2. Find Contours() oracle—an enchanter of pixels, a conjurer of lines. a spell, It weaves unraveling the mysteries of form. With every invocation, it whispers, “Here lies the essence of shape, the heartbeat of objects.” And so, the binary veil descends—a chiaroscuro of black and white. Objects stand stark against the void, their silhouettes etched in defiance. The contours, like ancient runes, encode the language of existence—a Lexicon of Curves, Corners, and Hidden Thresholds. In this cryptic dance, a leaf unfurls—a delicate arabesque against the morning dew. A cat’s silhouette prowls—a feline sonnet of grace and stealth. The contours of a distant mountain—Etched by Aeons, whispered by winds— beckon explorers and dreamers alike. But wait! These contours are no mere sketches; they are the architects of vision In motion detection, they orchestrate alarms—a waltz of change, a Tango of Intrusion. For unattended objects, they play sentinels—Vigilant Guardians of empty corridors, silent witnesses to the mundane and the mysterious. And when the world splits—foreground from background—the contours wield their scalpel. They dissect reality, revealing the mundane teacup from the cosmic nebula. They are the seamstresses of segmentation, stitching together fragments of perception into coherent narratives. Dear seeker of contours, remember this: they are not mere lines; they are whispers of existence. They map the boundaries of our universe, connecting pixels to purpose. They are the brushstrokes of visionaries, the breadcrumbs for algorithms hungry for meaning. So, when you gaze upon an image, squint your eyes, and see beyond pixels.

There, amidst the binary symphony, the contours await—a silent ballet of significance. And in their delicate embrace, the mundane becomes magical, the ordinary transcendent. For within contours lies the promise: that even in the pixelated tapestry, we find poetry—a dance of light, a roadmap to wonder. Figure 5(b) depicts the whole mechanism.

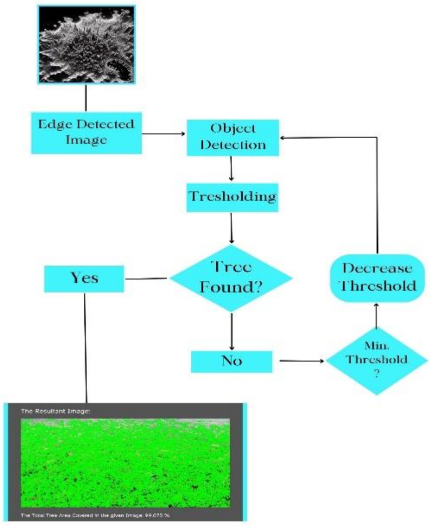


Fig. 5(b). Contour Detection Process

Algorithm 1: Canny Algorithm

Step 1: Read the input image, I, as a grayscale image.

Step 2: Blur the image using a Gaussian kernel for noise reduction.

Formula for Gaussian filter;

$$G(x,y) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Step 3: Calculate gradient strength and direction using Sobel filters.

Formula for Sobel operator:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix},$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\mathbf{G} = \sqrt{\mathbf{G}_x^{\,2} + \mathbf{G}_y^{\,2}}$$

Compute gradient magnitude:

Compute gradient direction: $\Theta = \operatorname{atan2}(\mathbf{G}_y, \mathbf{G}_x)$

Step 4: To sharpen edges, compare gradient magnitudes with neighbors and discard non-maximum pixels

Step 5: Apply double thresholding to Detect Potential Edges.

Define two thresholds:

- Low threshold: T_{Low}
- High threshold: T_{High}

Classification of pixels based on thresholds:

- Strong edges: $G > T_{High}$
- Non-Edges: $G < T_{Low}$
- Weak Edges: $T_{Low} \leq G \leq T_{High}$

Step 6: To sharpen edges, connect strong edges smoothly, monitor weak links, and merge them into the edge, Update robust edge set.

Step 7: The output of Canny edge detection is the resulting image after edge tracking, denoted as:

Output=EdgeTracking(I) ; I – Input Image

The contour detection technique finds edges first using Sobel operators and then recognises corners using Harris corner detection. Line segments are detected and their intersections analysed to find T-junctions. Contour points are combined, such as edges, corners, and T-Junctions. Contour-tracing algorithms follow the contours. Finally, closed contours are obtained by joining segments and filtering noise. This approach delineates object boundaries using gradient-based edge detection, corner identification, line detection, and contour-following algorithms, making computer vision tasks like object recognition and scene interpretation easier.

Algorithm 2: Contour Detection

Step 1: Read the Edge Detected Image given by Canny Algorithm.

Step 2: Compute gradients for I1, I2 with Sobel, then find Magnitude.

$$|\nabla I| = \sqrt{(I_x^2 + I_y^2)}.$$

Threshold to get binary edges.

Step 3: Harris Corner Detection

a. Construct structure tensor M:

$$M = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}.$$

b. Compute Harris response R for each pixel

$$R = det(M) - k \cdot trace(M)^{2}$$

c. Threshold R to find corners.

Step 4: Hough transform finds line segments $\{L1, L2, \ldots\ldots, Ln\}$

for y= mx+ c$\rho = x\cos(\theta) + y\sin(\theta)$

a. Locate intersecting point coordinates (x,y).
b. if N (x, y) ≥ 3:
count (intercept (x, y));

Step 5: Contour Extraction

i. Combine Edges, corners & T-junctions to form contour points:

$C = \{p_1, p_2, \ldots\ldots, p_n\};$

p- Contour Point,

C= Set of Contours.

ii. Utilizing Freeman chain code we obtain the sequence of Connected Contours:

$C' = \{p_1', p_2', \ldots\ldots, p_m'\};$

iii. Connect segments, filter noise for final contours, denoted as C":

$C'' = \{p_1'', p_2'', \ldots\ldots, p_m''\};$

Resultant Contours are drawn up as C”.

---

4. Results And Analysis

1. Case Study:

Assume Gaussian blur with a simple 3×3 averaging filter kernel. Proceed with remaining steps after discussing each in detail.

Input Image: $I_{3\times3}$ =

$$\begin{pmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{pmatrix}$$

Blur the Image Using Gaussian Kernel Applying this kernel to the input matrix for blurring:

Blurred Image =Convolution (I, Gaussian Kernel)

Let's compute the blurred image:

$$\text{Blurred Image} = \begin{pmatrix} (10+20+30)/9 & (20+30+40)/9 & (30+40+50)/9 \\ (60+70+80)/9 & (70+80+90)/9 & (80+90+100)/9 \\ (110+120+130)/9 & (120+130+140)/9 & (130+140+150)/9 \\ (160+170+180)/9 & (170+180+190)/9 & (180+190+200)/9 \\ (210+220+230)/9 & (220+230+240)/9 & (230+240+250)/9 \end{pmatrix}$$

$$= \begin{pmatrix} 20 & 30 & 40 \\ 70 & 80 & 90 \\ 120 & 130 & 140 \\ 170 & 180 & 190 \\ 220 & 230 & 240 \end{pmatrix}$$

Calculate Gradient Strength and Direction Using Sobel Filters We'll compute the gradient magnitudes G and directions θ using Sobel operators G1 & G2

Given the Sobel operators:

$$G_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, \quad G_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

We'll convolve the blurred image with these Sobel operators to compute G1 & G2 Let's denote G1 & G2 as matrices representing the convolutions of I with the Sobel operators

$$G_x = \begin{pmatrix} -10 & 0 & 10 \\ -20 & 0 & 20 \\ -10 & 0 & 10 \end{pmatrix}, \quad G_y = \begin{pmatrix} -10 & -20 & -10 \\ 0 & 0 & 0 \\ 10 & 20 & 10 \end{pmatrix}$$

Now we can compute the gradient magnitudes & directions:

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

$$\mathbf{\Theta} = \operatorname{atan2}(\mathbf{G}_y, \mathbf{G}_x)$$

$$G = \begin{pmatrix} 14.142 & 20 & 14.142 \\ 28.284 & 0 & 28.284 \\ 14.142 & 20 & 14.142 \end{pmatrix}$$

$$\theta = \begin{pmatrix} -135^\circ & -90^\circ & -45^\circ \\ -180^\circ & 0^\circ & 180^\circ \\ 135^\circ & 90^\circ & 45^\circ \end{pmatrix}$$

Using this we can compute:

Step 4: Non-maximum Suppression In this step, we discard non-maximum pixels by comparing gradient magnitudes with their neighbors and keeping only the maximum values along the gradient direction. Given the gradient magnitudes G and directions θ, we'll perform non-maximum suppression. Since we already computed G and θ earlier, let's assume that we have performed non-maximum suppression, and the resulting image is:

Step 5: Double Thresholding We have defined low threshold T_{Low} = 10 and high threshold T_{High}= 25. We'll classify pixels into strong edges, non-edges, and weak edges based on these thresholds.

Given the gradient magnitudes G, we'll classify pixels as:

i. Strong edges: G>25
ii. Non-edges: G<10
iii. Weak edges: 10≤G≤25

$$\text{Thresholded Image} = \begin{pmatrix} 0 & 0 & 0 \\ 255 & 0 & 255 \\ 0 & 0 & 0 \end{pmatrix}$$

Here, pixels with values greater than 25 are classified as strong edges (assigned value 255), while others are set to 0.

Step 6: Edge Tracking by Hysteresis

In this step, we'll connect strong edges smoothly and monitor weak links, merging weak edges with strong edges to form continuous edges. Since we don't have any weak edges in this simplified example, we can consider the thresholded image itself as the final output.

Step 7: The output of the Canny edge detection algorithm based on the provided input matrix and the simplified steps is the thresholded image:

$$\text{Thresholded Image} = \begin{pmatrix} 0 & 0 & 0 \\ 255 & 0 & 255 \\ 0 & 0 & 0 \end{pmatrix}$$

This matrix represents the detected edges after applying Canny edge detection.

Step 8: Computing the Gradient Matrices I_1 & I_2:

$$I1 = \begin{bmatrix} -1 & 0 & 1 \\ -255 & 0 & 255 \\ -1 & 0 & 1 \end{bmatrix}$$

$$I2 = \begin{bmatrix} 1 & 255 & 1 \\ 0 & 0 & 0 \\ -1 & -255 & -1 \end{bmatrix}$$

&

Step 9: Magnitude of gradients are as follows:

$$M = \begin{bmatrix} \sqrt{2} & 255 & \sqrt{2} \\ 255 & 0 & 255 \\ \sqrt{2} & 255 & \sqrt{2} \end{bmatrix}$$

Step 10: calculating the determinant and Trace of M:

$$\begin{aligned}\mathbf{det}(\boldsymbol{M}) = (\sqrt{2} \times 0 \times \sqrt{2}) + (255 \times 255 \times \sqrt{2}) \\ + (\sqrt{2} \times 255 \times 255) \\ + (\sqrt{2} \times 255 \times \sqrt{2}) \\ - (255 \times 0 \times \sqrt{2}) \\ - (\sqrt{2} \times 255 \times 255)\end{aligned}$$

$$\begin{aligned}\mathbf{det}(\boldsymbol{M}) = 0 + 65025\sqrt{2} + 65025\sqrt{2} - 0 - 0 \\ - 65025\sqrt{2}\end{aligned}$$

$$\mathbf{det}(\mathbf{M}) = 130050\sqrt{2}$$

Trace: $\boldsymbol{trace}(\boldsymbol{M}) = \sqrt{2} + 0 + \sqrt{2} = 2\sqrt{2}$

Now that we have these values, we can proceed with the Harris Corner Detection and Hough Transform steps if necessary.

Let's set k=0.04 (a common number in Harris Corner Detection). We can now compute R.

$R = 130050\sqrt{2} - 0.04 \times (2\sqrt{2})^2$

$R = 130050\sqrt{2} - 0.04 \times 8$

$R = 130050\sqrt{2} - 0.32$

$R = 1,83,890.38$

Step 11: Thresholding: Select a threshold value (T). Select pixels with R values higher than the threshold by comparing them to T.

$$\text{SelectedPixels}(x, y) = \begin{cases} 1, & \text{if } R(x, y) > T \\ 0, & \text{otherwise} \end{cases}$$

Non-Maximum Suppression: After thresholding, use non-maximum suppression to eliminate local maxima in the R response.

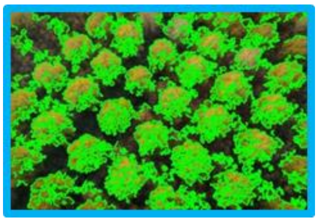
This step prevents neighboring pixels with comparable R values from being treated as different corners.

For each pixel:

$$\text{Suppress}(x, y) = \begin{cases} R(x, y), & \text{if } R(x, y) > \max(R(\text{neighborhood}(x, y))) \\ 0, & \text{otherwise} \end{cases}$$

Step 12: To convert edge points into Hough space, use the Hough accumulator to determine their associated parameters. Mathematically, the edge point (x, y) can be transformed into a parameter space (e.g., slope-intercept space, y=mx + c). To discover important lines, accumulate votes in the Hough space. Mathematically, each edge point in the Hough space accumulates votes for lines that pass through it. To improve identified lines, consider approaches such as line fitting or interpolation. Mathematically, refinement strategies might include fitting a model to collected votes or interpolating between observed line segments.

Output Image Obtained:



✓ *The Total Tree Area Covered in the given Image:*
93.6017706302794 %

2. *System Specifications:*

Results of the analysis are predicated on running a Python framework on a 64-bit version of Windows 11 using hardware specs such as an 11th Generation Intel® Coretm I5 Processor, 512 GB SSD storage, and 12 GB RAM. The technique facilitates environmental study by precisely quantifying the percentage of area covered by trees in input photos. The output's character count offers information about the extent of tree covering, supporting environmental monitoring and assessment initiatives.

3. *Performance Analysis:*

The responsiveness of our project's tree detection algorithm which includes the time needed for image processing and tree-cover detection determines how effective the method is. Numerous variables affect this response time, including as the system's processing power, the quantity and quality of the input images, hardware and software setup optimizations, and the algorithm's complexity. In real-time applications like urban planning and Environmental Monitoring, it is crucial to minimize the algorithm's reaction time to ensure its efficacy.

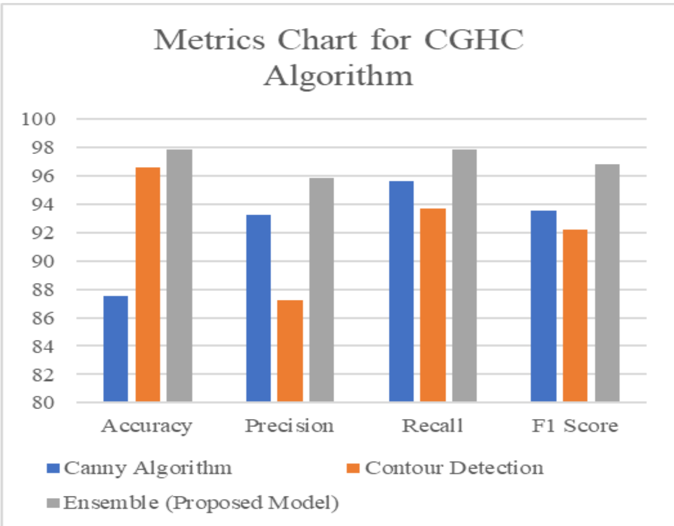


Table 3. Describing The Various Cases To Obtain Results

S.No	Input image	Output image	Percentage covered
1.			93.45%

2.			67.14%
3.			8.529%
4.			0%

In the images described in the Table 3, the first scenario showcases dense tree coverage, constituting 93.45% of the image, and is represented by a Green Mask. The second scenario exhibits moderate tree coverage, covering 67.14% of the image and marked with a Yellow Mask. In contrast, the third scenario features a single tree with timid coverage of 8.529%, possibly suitable for development, and is depicted with a Red-Colored Mask. However, the fourth scenario depicts a Clear Sky with 0% Tree Coverage, serving as a false test case. These scenarios demonstrate varying degrees of tree coverage, ranging from dense vegetation to sparse foliage and clear skies.

- Table 3 displays the metrics of each dived method, canny and contour, as well as the final recommended model metrics, with a summary.

Table 4. Metrics For Particular Algorithms

Algorithm	Accuracy	Precision	Recall	F1 Score
Canny Algorithm	87.54	93.24	95.65	93.56
Contour Detection	96.55	87.25	93.66	92.21
Ensemble (Proposed Model)	97.87	95.83	97.87	96.84

The table illustrates the evaluation results of the Canny and Contour algorithms utilized within the ensemble model. These algorithms play pivotal roles in image processing and feature extraction tasks. While both algorithms demonstrate high performance, there are discernible differences in their individual metrics. The Canny algorithm showcases slightly higher accuracy and F1 score, indicating robust performance in detecting edges and features. Conversely, the Contour algorithm exhibits marginally better precision and recall, suggesting superior object delineation capabilities. These nuanced distinctions highlight the complementary nature of the algorithms within the ensemble, contributing synergistically to the overall effectiveness of the image processing framework.

Fig 6. Bar Chart Representing Metric Values of Algorithms

5. Conclusion

Our project embodies an unparalleled vision, seamlessly merging environmental stewardship with sustainable development, thus birthing the groundbreaking ENV-DEVELOPMENT society. This groundbreaking concept stands as a rallying cry against the dire challenges of our era, from the looming threat of climate change to the relentless degradation of our natural habitats and finite resources. By interweaving environmental consciousness into the fabric of progress, we strive to cultivate a harmonious coexistence between human advancement and ecological preservation. Through State-of-The-Art solutions and transformative strategies, our initiative empowers global leaders, urban planners, and policymakers to carve out a future where sustainability reigns supreme. As our narrative unfolds, it becomes undeniably evident that our project is not just a response to modern-day crises but a guiding light towards a future defined by resilience, equity, and prosperity. Embracing the ethos of ENV-DEVELOPMENT, we navigate a path towards a brighter tomorrow, underscoring our indispensable role in shaping the trajectory of sustainable progress on a global scale, all while maintaining model Accuracy and Precision at an astounding 97.87% and 95.83%, respectively.

References

[1] Aluri, Yuva & Kanuri, Naveen & Krishna, Marlapalli & Busi, Rambabu & Chowdary, S & Rani, Bandlamudi. (2021). Framework for Tree Crown Delineation from Panchromatic Images. EAI Endorsed Transactions on Energy Web. 9. 169428. 10.4108/eai.29-4-2021.169428.

[2] Ertel, J., Goldman, L., Spore, J., & Brandt, J. (2023, June 6). Tree Cover Data Comparison. Global Forest Watch. Retrieved from https://www.globalforestwatch.org/blog/data-and-research/tree-cover-data-comparison.

[3] A. Khan et al., "Remote Sensing: An Automated Methodology for Olive Tree Detection and Counting in Satellite Images," in IEEE Access, vol. 6, pp. 77816-77828, 2018, doi: 10.1109/ACCESS.2018.2884199.

[4] M. Waleed, T. -W. Um, A. Khan and Z. Ahmad, "An Automated Method for Detection and Enumeration of Olive Trees Through Remote Sensing," in IEEE Access, vol. 8, pp. 108592-108601, 2020, doi: 10.1109/ACCESS.2020.2999078.

[5] P. Faezah, I. Asmida, M. H. Siti Khairiyah, J. S. Norrizah and C. A. Nuraini, "Diversity and tree species community at Bukit Nanas Forest Reserve, Kuala Lumpur," *2013 IEEE Business Engineering and Industrial Applications Colloquium (BEIAC)*, Langkawi, Malaysia, 2013, pp. 846-850, doi: 10.1109/BEIAC.2013.6560255.

[6] C. Lin, C. -S. Lo and G. Thomson, "A textural modification of the MMAC algorithm for individual tree delineation in forest stand using aerial bitmap images," *2011 4th International Congress on Image and Signal Processing*, Shanghai, 2011, pp. 1604-1608, doi: 10.1109/CISP.2011.6100501.

[7] C. Yu, M. Li and M. Zhang, "Classification of dominant tree species in an urban forest park using the remote sensing image of WorldView-2," 2015 8th International Congress on Image and Signal Processing (CISP), Shenyang, China, 2015, pp. 742-747, doi: 10.1109/CISP.2015.7407976.

[8] Li, H.; Hu, B.; Li, Q.; Jing, L. CNN-Based Individual Tree Species Classification Using High-Resolution Satellite Imagery and Airborne LiDAR Data. Forests 2021, 12, 1697. https://doi.org/10.3390/f12121697 2021.

[9] Kim, D.-H.; Ko, C.-U.; Kim, D.-G.; Kang, J.-T.; Park, J.-M.; Cho, H.-J. Automated Segmentation of Individual Tree Structures Using Deep Learning over LiDAR Point Cloud Data. Forests 2023, 14, 1159. https://doi.org/10.3390/f14061159 2023.

[10] Li, D.; Shen, X.; Yu, Y.; Guan, H.; Li, J.; Zhang, G.; Li, D. Building Extraction from Airborne Multi-Spectral LiDAR Point Clouds Based

on Graph Geometric Moments Convolutional Neural Networks. Remote Sens. 2020, 12, 3186. https://doi.org/10.3390/rs12193186 2020.

[11] J. Yang, Y. He and J. Caspersen, "Individual tree-based species classification for uneven-aged, mixed-deciduous forests using multi-seasonal WorldView-3 images," *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Fort Worth, TX, USA, 2017, pp. 827-830, doi:10.1109/IGARSS.2017.8127080.

[12] Kandula, Ashok Reddy, R. Sathya, And S. Narayana. "Comparative Analysis Of Machine Learning Techniques On Genetic Mutation-Based Cancer Diagnosis Data." Journal Of Theoretical And Applied Information Technology 100.6 (2022) https://api.semanticscholar.org/CorpusID:265675372.

[13] Hou, C.; Liu, Z.; Chen, Y.; Wang, S.; Liu, A. Tree Species Classification from Airborne Hyperspectral Images Using Spatial–Spectral Network. Remote Sens. 2023, 15, 5679. https://doi.org/10.3390/rs15245679 2023.

[14] Kandula, Ashok Reddy, R. Sathya, and S. Narayana. "Multivariate Analysis on Personalized Cancer Data using a Hybrid Classification Model using Voting Classifier." International Journal of Intelligent Systems and Applications in Engineering 11.1 (2023): 354-362 https://ijisae.org/index.php/IJISAE/article/view/2546.

[15] Tanya V. Olegario; Renann G. Baldovino; Nilo T. Bugtai; (2020). A Decision Tree-based

Classification of Diseased Pine and Oak Trees Using Satellite Imagery. 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), (), – doi:10.1109/hnicem51456.2020.9400002 .

[16] A. Gupta, J. Byrne, D. Moloney, S. Watson and H. Yin, "Tree Annotations in LiDAR Data Using Point Densities and Convolutional Neural Networks," in IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 2, pp. 971-981, Feb. 2020, doi: 10.1109/TGRS.2019.2942201. using Deep Learning Techniques to Identify Pneumonia," 2023 Second International Conference on Augmented Intelligence and

[17] A. Reddy Kandula, S. D. Gangiredla, K. P. Tirukkovalluri, S. Yallamilli, N. Tummalapalli and S. Talluri, "X-Ray Image Analysis Sustainable Systems (ICAISS), Trichy, India, 2023, pp. 674-679, doi: 10.1109/ICAISS58487.2023.10250543.

[18] K. Yilong, T. Junsnan, Z. Huaiqmg, J. Xian, L. Kangnmg and Z. Ning, "Research on forest visualization system based on data cube," 2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE), Zhangjiajie, China, 2012, pp. 769-773, doi: 10.1109/CSAE.2012.6272879.

[19] A. R. Kandula, S. Kalyanapu, L. S. Gottipati, H. Kamma, M. Munagala and P. Ramachandran, "Revolutionizing Malaria Diagnosis: A Deep Learning Approach," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 63-66, doi: 10.1109/ICSCSS57650.2023.10169586.

[20] D. I. Moody et al., "Land cover classification in fused multisensor multispectral satellite imagery," 2016 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI), Santa Fe, NM, USA, 2016, pp. 85-88, doi: 10.1109/SSIAI.2016.7459181.

[21] S. Kalyanapu, A. R. Kandula, P. Madhuri, P. Ganta, H. N. Sai Sri Vattikuti and U. Darapu, "Enhancing Breast Cancer Prediction Through SVM-Based Analysis," 2023 Annual International Conference on Emerging Research Areas: International Conference on Intelligent Systems (AICERA/ICIS), Kanjirapally, India, 2023, pp. 1-6, doi: 10.1109/AICERA/ICIS59538.2023.10420106.

[22] E. -s. Kim, J. -s. Lee, G. -e. Park and J. -h. Lim, "Change Analysis of Subalpine Coniferous Forest Area over the Last 20 Years using Time-series Landsat Images," IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 2018, pp. 5240-5242, doi: 10.1109/IGARSS.2018.8519000.

[23] Q. Tan and J. Wang, "Mapping urban tree coverage using object-oriented image analysis method: A case study," 2009 IEEE International Geoscience and Remote Sensing Symposium, Cape Town, South Africa, 2009, pp. III-342-III-345, doi: 10.1109/IGARSS.2009.5417772.

[24] W. Huang, H. Li and G. Lin, "Classifying forest stands based on multi-scale structure features using Quickbird image," 2015 2nd IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services (ICSDM), Fuzhou, China, 2015, pp. 202-208, doi: 10.1109/ICSDM.2015.7298054.

[25] T. V. Olegario, R. G. Baldovino and N. T. Bugtai, "A Decision Tree-based Classification of Diseased Pine and Oak Trees Using Satellite Imagery," 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Manila, Philippines, 2020, pp. 1-4, doi: 10.1109/HNICEM51456.2020.9400002.

[26] A. R. Kandula, M. Tadiparthi, P. Yakkala, S. Pasupuleti, P. Pagolu and S. M. Chandrika Potharlanka, "Design and Implementation of a Chatbot for Automated Legal Assistance using Natural Language Processing and Machine Learning," 2023 Annual International Conference on Emerging Research Areas: International Conference on Intelligent Systems (AICERA/ICIS), Kanjirapally, India, 2023, pp. 1-6, doi: 10.1109/AICERA/ICIS59538.2023.10420298