UNIVERSITI PUTRA MALAYSIA

DETECTION OF OIL PALM FRONDS USING VISION SYSTEM

DIANA SABRYNA SIRAH ANAK SABANG

DETECTION OF OIL PALM FRONDS USING VISION SYSTEM

DIANA SABRYNA SIRAH ANAK SABANG

181578

BACHELOR OF AGRICULTURAL AND BIOSYSTEMS ENGINEERING

FACULTY OF ENGINEERING

UNIVERSITY PUTRA MALAYSIA

SERDANG, SELANGOR

2018/20019

PROJECT APPROVAL LETTER

This project report attached here, entitled **"DETECTION OF OIL PALM FRONDS USING VISION SYSTEM"** prepared and submitted by **DIANA SABRYNA SIRAH ANAK SABANG** in partial fulfillment of the requirement for Bachelor of Agricultural and Biosystems Engineering is hereby accepted.

____

(Dr. Mahirah Bt. Jahari) Date:

Project Supervisor

Department Of Biological And Agricultural Engineering

Faculty Of Engineering, University Putra Malaysia

____

(Assoc. Prof. Dr. Siti Khairunniza bt. Bejo) Date:

Panel Examiner

Department Of Biological And Agricultural Engineering

Faculty Of Engineering, University Putra Malaysia

____

(Dr. Nurulhuda bt. Khairuddin) Date:

Panel Examiner

Department Of Biological And Agricultural Engineering

Faculty Of Engineering, University Putra Malaysia

DECLARATION OF THESIS AND COPYRIGHT

The Library,

University Putra Malaysia,

43400 Serdang,

Selangor Darul Ehsan.

<u>PERMISSION TO MAKE PHOTOCOPIES OF REPORT AND THESIS</u>

1. DIANA SABRYNA SIRAH ANAK SABANG declares that the project report for thesis entitled **"DETECTION OF OIL PALM FRONDS USING VISION SYSTEM"** belongs to my supervisor Dr. Mahirah Bt. Jahari and me. The content of report may be used by anyone for the academic purpose of teaching, learning and research only. Therefore, Universiti Putra Malaysia is permitted to make copies of document for the same academic purpose.

Date :

Signature :

Name :

ACKNOWLEDGEMENT

No one walks alone on the journey of life. So first and foremost, I would like to give thanks to God for protection and ability to do work. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of my final year project. Apart from the efforts of I have put on, the success of any project depends largely on the encouragement and guidelines of my supervisor, Dr. Mahirah Bt. Jahari for the golden opportunity in doing the research for “Detection of Oil Palm Fronds Using Vision System”.

Secondly, an honorable mention goes to our families and friends for their understandings and supports. The guidance and support received from all the members who contributed and who are contributing to this project, was vital for the success of the project. I am grateful for their constant support and help.

Finally all countless credit to the writer, publisher and all people include in the references that I have used. Without reference of their work, this study cannot be done professionally.

ABSTRACT

Oil palm plantations in Malaysia still use traditional method to prune the fronds. Even with innovation in mechanization and automation, the created oil palm fronds cutter still relies on human vision. As the country's technology is growing, this creates the idea of vision system in line with human vision. This image recognition system has relation to image processing. Edge detection algorithm is useful because it helps to create the object's perimeter. This paper emphasize on analyzing RGB image using edge detection and some other morphological parameters. Developed algorithm is analyzed for its performance and will be enhance in future.

ABSTRAK

Perladangan kelapa sawit di Malaysia masih menggunakan kaedah tradisional untuk memangkas lepah kelapa sawit. Walaupun ada inovasi dalam mekanisasi dan automasi, pemotong lepah kelapa sawit yang dicipta masih bergantung kepada penglihatan manusia. Memandangkan teknologi dalam negara semakin berkembang, ini menimbulkan idea untuk mewujudkan sistem penglihatan yang selaras dengan penglihatan manusia. Sistem pengenalan imej ini saling berkait dengan pemprosesan imej. Algoritma pengesanan tepi berguna kerana ia membantu untuk membuat perimeter objek. Kajian penulisan ini memberi penekanan untuk menganalisis imej RGB menggunakan pengesanan pinggir dan beberapa parameter morfologi lain. Algoritma yang dicipta akan dianalisis untuk prestasi dan dipertingkatkan pada masa akan datang.

Table of Content

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATION

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 General

Compared to the rest of the country in the world, Malaysia is geographically small but it is an arable land for new oil palm plantations. However, Malaysia is currently the world largest exporter of palm oil. As the producer of palm oil, Malaysia is the second ranked country compared to Indonesia (REUTERS, 2018). According to Choo (2012), the palm industry has great opportunity related to nutrition caused by the demand in food industries due to its health benefits.

The large capacities of the oil production are very much related to the good agricultural practices, good processing and harvesting methods. Regarding the technique, most of the plantation in Malaysia still uses the traditional method to harvest the Fresh Fruit Bunch (FFB) and tree pruning. For the short tree, they use chisel while for the tall tree, sickle is used. For longer tree of above 3m, they attached a long pole with the sickle.

In order for Malaysia's agricultural sector to move forward, adopting smart farming or precision agriculture is very much recommended. The smart farming, also known as the Third Green Revolution applies technology related to information and communications in agriculture which includes include the use of drones to carry out monitoring of crops, planting, soil and field analysis, and crop spraying (Jaabi, 2017). The first step for achieving this target is to raise awareness among farmers of the benefits of new farming technology.

Other than that, precision farming also offer the application of vision system technology. It is widely applied in various agricultural exploitations along with the technology of image processing, graphics and rapid development of computer technology.

1.2 Significance of Study

The finding of this study will improve the method of cutting fronds and displays the important role of vision system in agriculture today. The great demand for sustainable agriculture will highlight the function of the vision system. Thus, new theory on vision system can be developed further for improvising in precision farming.

1.3 Problem Statement

Smart farming cannot be developed further if the current practice in Malaysia involves the use of traditional method, especially using human power. Traditional methods are outdated. This creates the need to use successful automation for the fronds cutting process that requires a vision system in line with human grading. Using vision system, the length of the fronds to be cut can be estimated using developed algorithm and the installed cutter on the Oil Palm Robotic Climber can easily cut the fronds.

1.4 Objectives

The main objective of this study is to detect the oil palm fronds using vision system. Stated below are the specific objectives to be achieved:

1. To develop an algorithm to detect oil palm fronds using image processing.
2. To evaluate the classification performance of the algorithm.

1.5 Scope and Limitation of Study

The study focuses on performing image analysis captured by the vision system and identify the different object in the image especially for oil palm fronds, FFB and oil palm sheath detection for the ease of detecting the oil palm fronds. The heights of the trees are below 10m.

CHAPTER 2

LITERATURE REVIEW

2.1 Computer Vision System

Vision systems are structured into two main components: camera and lighting device. Lenses are very important accessories for vision systems. Every camera need a lens with certain properties in order to choose the appropriate field if view and to image the object in clear focus to the sensor of the camera. To record a good image for desired object, light source are needed for illumination. Nixon and Aguado (2012) explain that basically, a computer vision system requires a computer, camera and the camera interface. The images acquired from camera are processed using some computer software.

For human, the most powerful image processing equipment known as a vision system is the brain. Computer vision tries to mimic human vision. The input is the images and the output will be knowledge of the scene. It studies an image or a group of images, use image processing and machine learning techniques, to mine information from image other than its properties.

Any visual input will be understood and predicted. Kaiser (2017) write that the ultimate goal is to use computers to duplicate human vision by processing the image acquisition, image processing and last but not least image analysis and understanding. To simplify our understanding, computer vision is a subset of image processing.

Computer vision is in the family of smart farming. Therefore, this includes the application in agriculture. It has various applications such as in the mature FFB image analysis, robotics, remote sensing and geoscience.

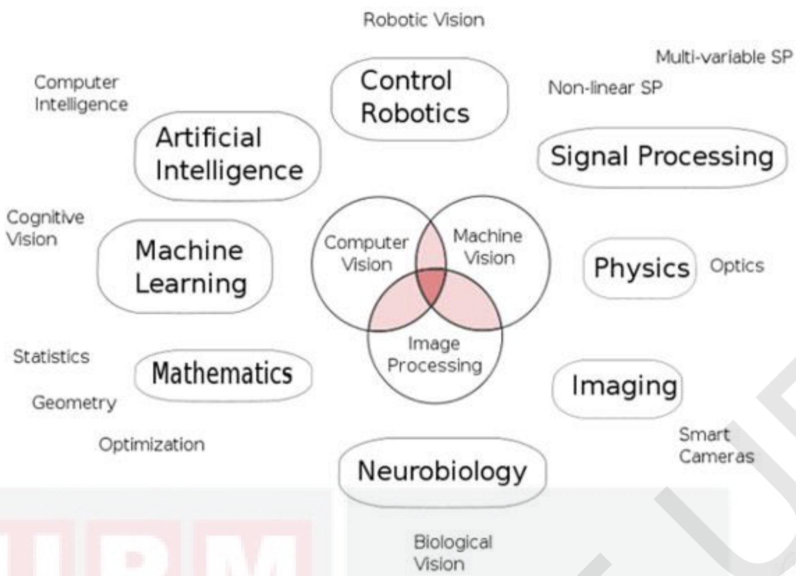


Figure 1: Computer Vision a field of Artificial Intelligence and Computer Science

2.1.1 Camera

Camera or known as the optical device is the second eyes for human that can capture and record images. Before buying camera, one needs to know the type camera to be used and function that it serves. Among the most used camera is the Digital Single Lens Reflex (DSLR). In blogs written by Shaw Academy (2016), they mention that the most important component in DSLR is the aperture (lens), the shutter (camera) and the light sensor (inside the camera).

DSLR use interchangeable lenses. Image taken by this camera is based on the type of lens attached to the camera. Aperture is that opening in a lens which allows the light in. The aperture is adjustable in size. That means we can increase and decrease the amount of light it lets in. The aperture also controls depth of field. An opening in lens is known as aperture. They control the amount of light in and controls depth of field. The light amounts are inversely proportional to the depth of field. The bigger the aperture, the more amount of light can enter but the depth of field is shallower and vice versa. For bigger aperture, it only focuses on the subject, the background and the foreground will be out of focus.

The specific opening time for shutter controls the light amount falling on the sensor (Moore, 2012). However, the amount of light fall on the camera sensor depends on the work done by the shutter and aperture.

2.1.2 Image Processing

Britannica (2014) define image processing as the work of computational techniques to analyze, enhance, compress, and reconstruct images. In short, it transforms the image. Both input and output of the processing are images. The input can be obtained by digital image. It analyzes and manipulates the image using specialized coding software applications.

Among the famous method for shape detecting in image processing is the edge detection technique. In studies conducted by Maini and Aggarwal (2009), it can identify and locate sharp discontinuities in an image. Enormous operators available for edge detection had been designed for specific type of edges. However, noisy image is burdensome as there is high frequency content in both the noise and the edges present in the image.

2.2 Oil Palm

2.2.1 History of Oil Palm

Oil palm or the scientific name Elaeis guineensis, was originated from West Africa. In conjunction with the British Industrial Revolution, this tropical perennial crop has proven its value in the international market especially with its output of palm oil. Other uses include the making of cosmetics, candle and industrial lubricants. Increase in palm oil demand had made the Europeans to invest the palm oil production to Southeast Asia. At 1870, it exists in Malaysia as an ornamental plant. Since Malaysia is a tropical country, the oil palm tree grows well. The location of the first commercial scale plantation was at the Tennamaran Estate in Selangor, 1917. (Source: The Oil Palm Blog, 2015)

Figure 2: Oil palm (Elaeis guineensis) in Universiti Putra Malaysia

2.2.2 Cutting of Oil Palm Fronds

The economic life of oil palm can lasts for around 25 to 30 years before it yields unprofitable output level. As for the Fresh Fruit Bunch (FFB) harvesting, it started 30 months after planting. Before harvesting the FFB, the fronds needs to removed either manually or mechanically. Besides for FFB harvesting purpose, the fronds still needs to undergo the trimming maintenance. According to Ycw, P. B. (2013), the oil palm tree bears a single vegetative shoot since it is single-stemmed palm. Therefore, they will continuously produce new leaf fortnight period. New leaves grow in young palms compared to the old palms. The writers also added that the tree's trunks will get smoother when they are older.

Based on Akvopedia (2018), the leaves grow in spiral of eight whereby the young leaves will grow on top of the tree while the older leaves will be at the bottom. Besides easing the FFB harvesting, the importance of cutting the fronds are to make sure the palm absorb optimum amount of sunlight, eliminate nutrients waste taken by unproductive fronds and to speed up the decomposition of pruned fronds. When pruning, optimum number of fronds should be left on the oil palm tree and it should focus on the dead fronds in palms less than 4 years after planting. It should be done before peak production and only during the dry period.

Table 1: Number of oil palm fronds to be cut based on year of plantation

Age of Oil Palm Tree (Year)	Number of Pruned Frond
5–7	48–56 fronds/palm (or: 2–3 fronds below the last ripe bunch)
8–15	40–48 fronds/palm (or: 1–2 fronds below the last ripe bunch)
15 and above	40 fronds/palm (or: 1 frond below the last ripe bunch)

2.2.3 Oil Palm Fronds Cutting Efforts by Universiti Putra Malaysia (UPM)

Returning to the traditional method of cutting the fronds using pole and high quality sickle, new improvement on the manually harvesting technique had been created by Associate Professor Ir. Dr. Norhisham Misron (Researcher at Faculty of Engineering UPM) and Abdul Razak Jelani (Malaysian Palm Oil Board Researcher Officer) known as E-cutter. Zakaria Azman (2016) writes that the technology development are using combined electric and motorized generator namely the A Double Standard Generator (high power density). It is petrol generated and weighing less than 9kg. The cutter vibrates in cutting the fronds and FFB. It can reach oil palm tree as high as 30feet.

Among the other available technology for FFB harvesting machines are scissor cutting mechanism, circular blade cutting mechanism, wire cutting mechanism mounted on tractor, cutting frond with wire cutting mechanism, MPOB-UPM's wheel-type mechanical harvesting machine, track-type harvesting machine for tall trees and climbing robot with tilt sensors which also known as Oil Palm Robotic Climber (Sharence Nai Sowat et al., 2018).

CHAPTER 3

METHODOLOGY

3.1 Sample Collection

3.1.1 Area of Study

The study was conducted at Taman Pertanian Universiti (TPU), University Putra Malaysia (UPM) oil palm plantation. Below is the information for the oil palm plantation:

Common Name	: Oil palm
Scientific Name	: Elaeis guineensis
Plantation Area	: 30ha
Distance Between Tree	: 8.8m x 8.8m x 8.8m
Date of Plantation	: May 2012

3.1.2 Equipment and Technique Used

In order to capture the image, one of the cameras available in the spatial lab was used. The selected camera to perform this operation was Canon EOS 400D. It is an entry-level digital single-lens reflex camera introduced by Canon on 24 August 2006. Information on Canon EOS 400D camera is provided below:

Sensor Resolution	: 10.1 Megapixel
Optical Sensor Type	: Complementary Metal Oxide Semiconductor (CMOS)
Image Recording Format	: RAW, JPEG, RAW + JPEG
Min. Operating Temperature	: 32 °F / 0^{o}C
Max. Operating Temperature	: 104 °F / 40^{o}C
Min. Focal Length	: 18 mm
Max. Focal Length	: 55 mm

Analyzing Selected Tree

10 random oil palm trees with the height below 10m, planted at flat ground are selected randomly. The tagging is made with numbering of 1 to 10. The tree height is measured using clinometer borrowed from Faculty of Forestry. Distance between the tree and observer are measured using measuring tape of 5m (16ft) length.

The equation to find height of tree is:

$$Tree\ height\ =\ Tan\ (Angle)^0 \times Distance\ from\ tree\ (m)$$
$$+\ Height\ from\ ground (m)$$

Sample calculation:

$$Tree\ 1\ = (Tan 17^o)\ \times 4.4m\ + 1.5m$$
$$= 2.8452\ m$$

Table 2: Number of tree sample and height of tree (m)

No. of Tree	Height of Tree (m)
1	2.8542
2	2.5158
3	3.0150
4	3.0150
5	2.7617
6	2.5970
7	3.1015
8	2.3552
9	3.1015
10	2.5158

Both the distance between tree and camera along with height of camera held is 1.5 m because when the image is too close we cannot see the FFB, oil palm fronds and the oil palm fiber properly due to the limitation of the camera view. The height of camera held depends on height of camera man. Each tree will have 4 images at 4 different angle namely angle A, B, C and D.

Table 3: Angles of camera for each oil palm tree

Name of angle	**Approximate value of angle**
A	0^{o} / 360^{o}
B	90^{o}
C	180^{o}
D	270^{o}

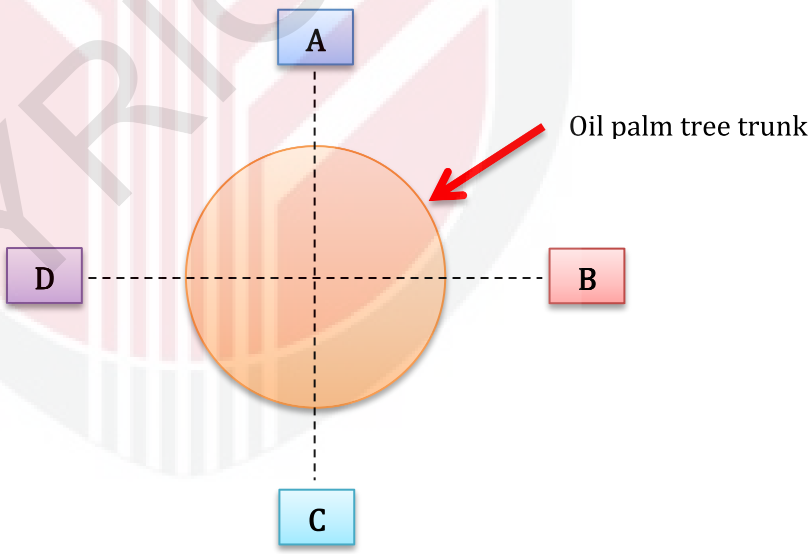


Figure 3: Angle of image acquisition from tree

3.1.3 Obtaining Image

The date was 8th of March 2019 on Friday morning with the day time of 0957–1047 hours. The weather condition is sunny but partly cloudy and the temperature range from 30^{o}C to 33^{o}C.

All the image shares the same properties as described below:

Image size (Dimension)	: 3888 x 2592
Color bit depth	: 24
F–number (F-stop)	: 5.6
Focal length	: 25mm
Shutter speed (Exposure Time)	: 1/30 sec
ISO Speed	: 125

The total samples from this shot are 40 images. The type of camera used is the Canon EOS 400D with the pixel resolution of 10.1 Megapixel. The image file format will be stored as Joint Photographic Experts Group (JPEG) format and transferred to computer.

3.2 Image Processing

MATLAB will used to find properties of the object. Among the function of it are to perform the analysis of the image the shapes, detecting edges, removing noise, counting objects, measuring the region and some other image processing techniques.

Method 1

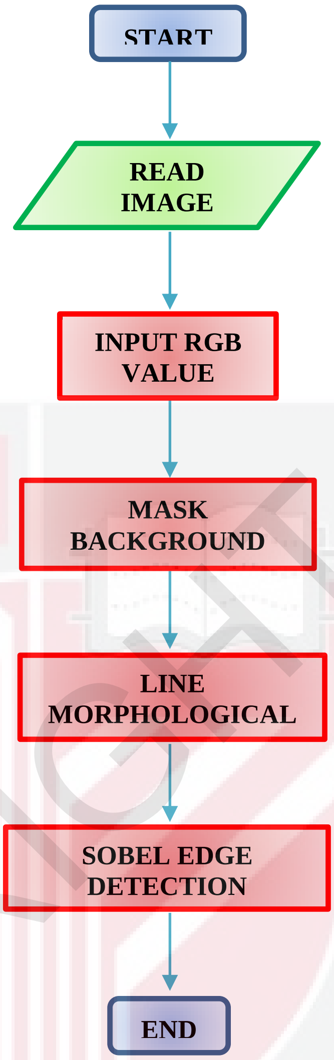


Figure 4: Method 1 flowchart diagram for the image processing procedure.

Method 2

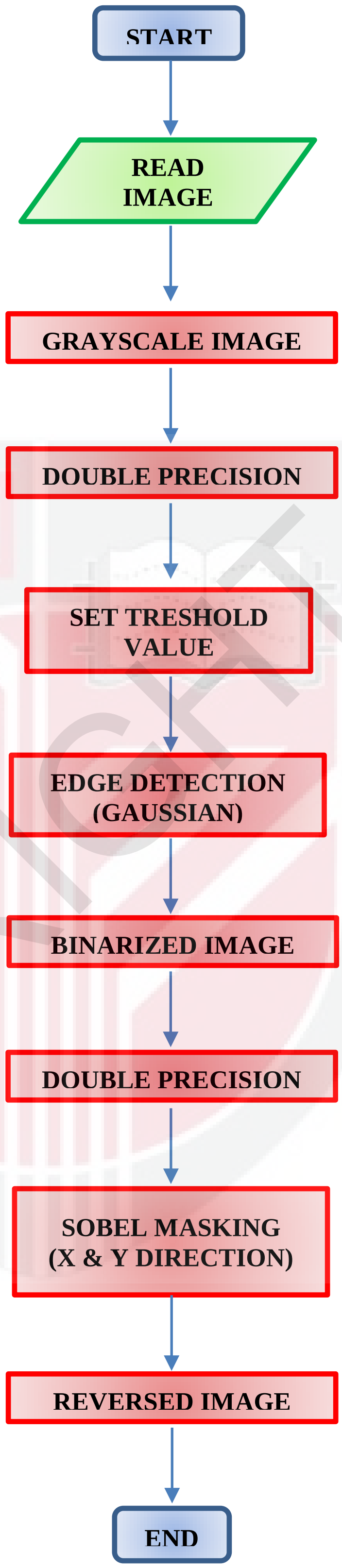


Figure 5: Method 2 flowchart diagram for the image processing procedure.

Method 3

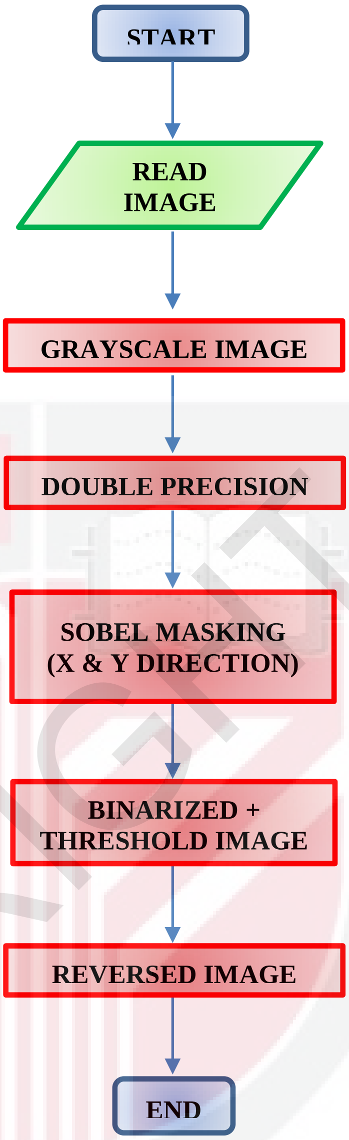


Figure 6: Method 3 flowchart diagram for the image processing procedure.

3.3 Statistical Analysis

An algorithm and the method for detecting the mathematically well-defined shapes will be developed using the MATLAB programming software. The role of computer vision in shape identification will yield many parameters. After some data collection has been made, an inference needs to be drawn through statistical analysis. When developing the algorithm, the shape needs to be described and represented based on the shapes available in the image for better detection and template comparison. The validity of the result depends on the developed algorithm.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Result

4.1.1 Method 1

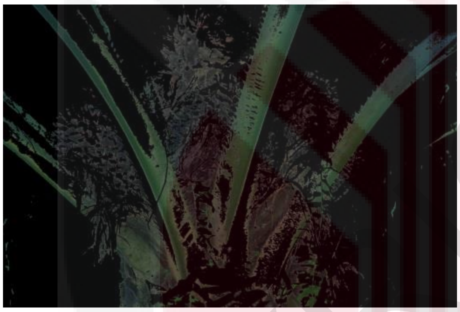
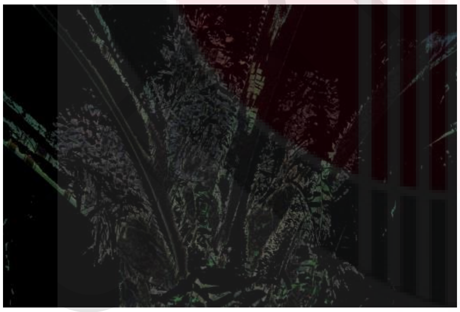
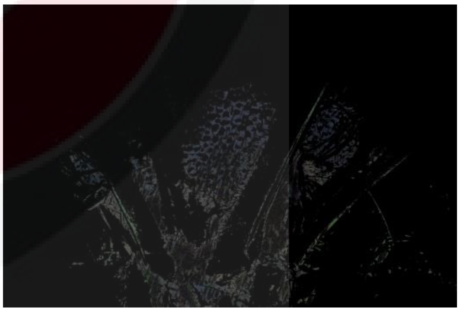
PRO’S	CON’S
	
1. Original image	1. Original image
	
2. Masked green background	2. Masked green background
	
3. Strel by length and degree	3. Strel by length and degree
	
4. Final image output	4. Final image output

Table 4: Image processing result for Method 1

4.1.2 Method 2

PRO'S

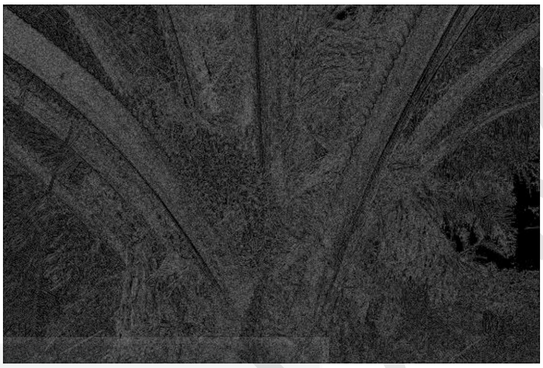
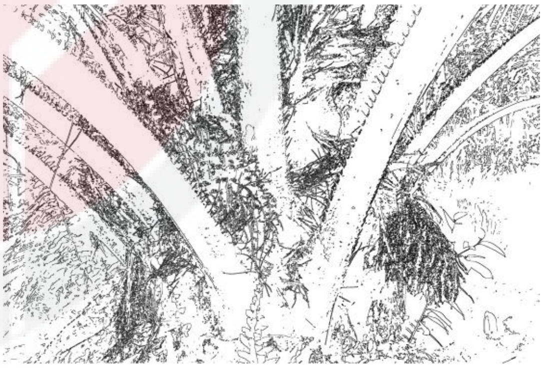
	
1. Original image	2. Binarized image
	
3. Threshold binarized image	4. Strel by length and degree
	
5. Sobel filter	6. Final image output

Table 5: Image processing result for Method 2 (Pro's)

CON'S

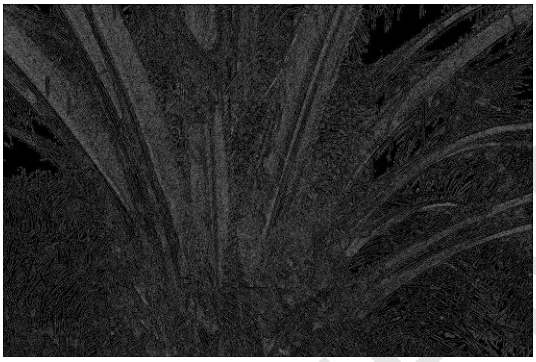
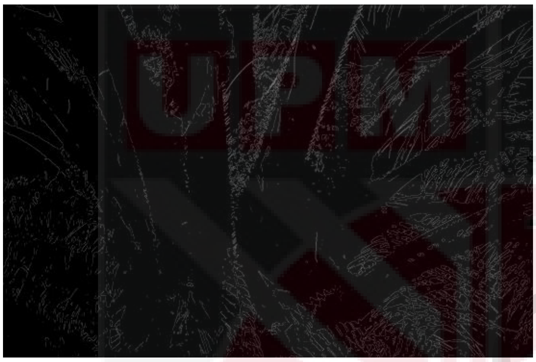
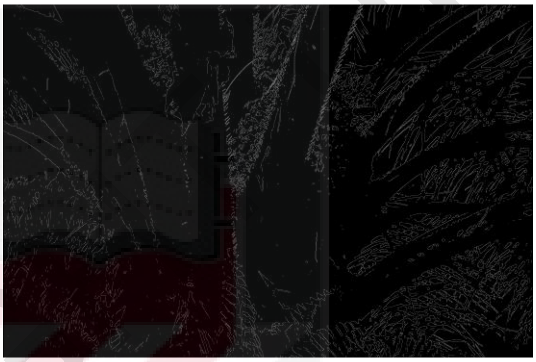
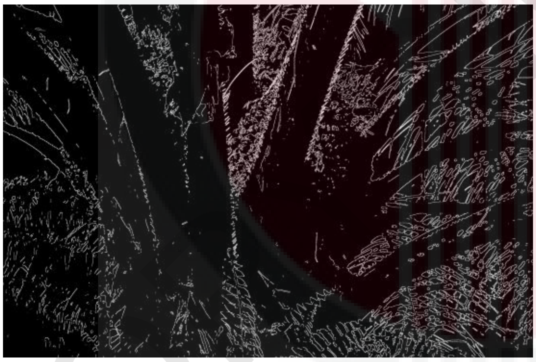
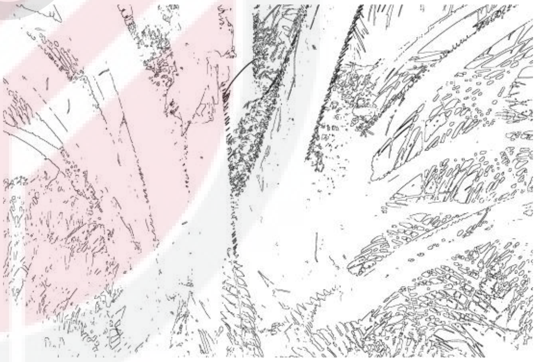
	
1. Original image	2. Binarized image
	
3. Threshold binarized image	4. Strel by length and degree
	
5. Sobel filter	6. Final image output

Table 6: Image processing result for Method 2 (Con's)

4.1.3 Method 3

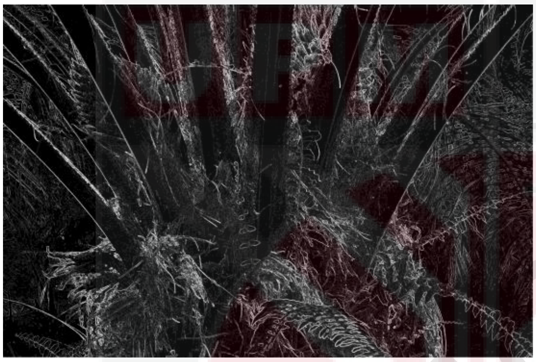
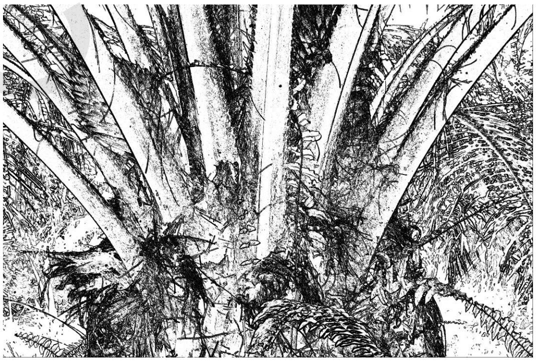
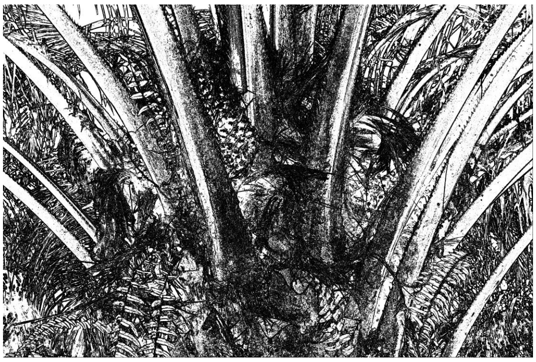
PRO'S	CON'S
	
1. Original image	
	
2. Sobel filter	
	
3. Binarized + Treshold output	
	
4. Reversed image	

Table 7: Image processing result for Method 3

4.2 Discussion

4.2.1 Method 1

Advantage (Pro's)

In Table 4 referring to the original image, the other oil palm tree leaves are considered as noise. This method focuses on eliminating the background green color and just portrayed the tree captured in the image. It can successfully help to eliminate the background and weeds growing on tree. It also focuses on the tree in the captured image as it highlights the fronds color section. Masked image can make us easily differentiate the shape of fronds, tree sheaths and FFB. Strel method by length and degree does help to detect the vertical lines accurately. After binarized, only the focused tree image left.

Limitation (Con's)

In the plantation, green is the dominant color in this image. Since the set RedGreenBlue (RGB) channel color of the fronds are also green, it also masked off the fronds green color. It makes some of the fronds part missing besides due to the masked growing weeds on tree. In some image, it eliminates the green color completely. RGB values are hard to set as green color intensity differs on every section. It detects the FFB better than the fronds part as the FFB color is not dominant.

4.2.2 Method 2

Advantage (Pro's)

For Table 5, the noises at background managed to be reduced until the main tree can be focused. The reduced threshold value of the binarized image gives the fronds segment easier to detect. After strel by lines and degrees, it is filtered with Sobel method for better edge detection. Method 2 can produce better definition for shape of oil palm fronds and FFB. In most images, the vertical lines of the oil palm fronds can be detected.

Limitation (Con's)

However in Table 6, due to set threshold value, some lines are disconnected and the fronds part attached to the tree cannot be determined. Most of the image attached on the appendices experience the same limitation. This means that the method depends on the light intensity. Brighter image makes the oil palm fronds edges harder to detect.

4.2.3 Method 3

Advantage (Pro's)

Based on result in Table 7, the Sobel method makes the edges become more prominent. Filtering the image by using x and y direction can detect shapes and edged very well which enabled us to see the shapes of oil palm fronds, FFB and tree. In most images attached at appendices, the fronds can be seen clearly. Since it displays the shapes very well, this method is ideal to be furthered for the vision system of the fronds cutter machine.

Limitation (Con's)

The noise at background cannot be removed fully even after thresholding the binarized image. Some image has messy outcome.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

All algorithms developed in the three selected methods can detect the main focused object in the picture, which are the oil palm fronds. It can be seen that they manage to display the shape and pattern which gave a good display for oil palm fronds, FFB, sheath and weeds that grows on the tree. Method 1 is not ideal as it also mask off the green color on the fronds causing it to be incapable reaching the objectives of this work. Method 2 can detect the fronds well but some of the lines are disconnected due to the set threshold value that has been set to the ideal value for all image. Method 3 can gave a fine image output especially on the fronds section as it can provide initial growing parts of the fronds itself.

5.2 Recommendation

Method 3 is the most ideal method for this project purpose as it gave a clear image of fronds. This method can be upgraded by highlighting the vertical lines and background of the non-straight line to be more transparent. Once improved, the cutting lines can be estimated for the cutter to precisely cutting the oil palm fronds. Area segmentation is not supported for fronds detections as every frond on every tree differ in size because not all of them will be pruned as the pruning has some parameters to be considered.

REFERENCE

1. REUTERS. (2018, January 18). Global palm oil output growth seen slowing in 2018 -analyst Fry. Retrieved from https://uk.reuters.com/article/malaysia-palmoil-outlook/global-palm-oil-output-growth-seen-slowing-in-2018-analyst-fry-idUKL3N1PC3E9

2. Choo, Yuen May. (2012, September). Malaysia: economic transformation advances oil palm industry. The American Oil Chemists' Society. Retrieved from https://www.aocs.org/stay-informed/read-inform/featured-articles/malaysia-economic-transformation-advances-oil-palm-industry-september-2012

3. Jaabi, Ainun. (2017, 5 September). Smart farming is the way to go. NEW STRAITS TIMES. Retrieved from https://www.nst.com.my/opinion/columnists/2017/09/276234/smart-farming-way-go

4. Nixon, Mark S.., Aguado, Alberto S.. (2012). *Feature Extraction & Image Processing for Computer Vision.* (3rd ed.). UK, London: Elsevier.

5. Kaiser, Adrien. (2017, January 12). What is Computer Vision? Retrieved from https://hayo.io/computer-vision/

6. Shaw Academy. (2016, February 19). Understanding a DSLR: How Does it Work?. Retrieved from https://www.shawacademy.com/blog/understanding-dslrs-how-does-a-dslr-work/

7. Moore, M. (2012, April 04). HOW DOES A DSLR WORK? Retrieved from https://martinmoorephotography.wordpress.com/2012/04/04/how-does-a-dslr-work/

8. Britannica, T. E. (2014, October 24). Image processing. Retrieved from https://www.britannica.com/technology/image-processing

9. Maini, R., & Aggarwal, H. (2009, February). Study and Comparison of Various Image Edge Detection Techniques. *International Journal of Image Processing (IJIP), 3*(1), 1-11. Retrieved December 2, 2018, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.301.927&rep=rep1&type=pdf

10. The Oil Palm. (2015, July 24). History and Origin. Retrieved from http://theoilpalm.org/history-and-origin/

11. Ycw, P. B. (2013, December 15). The Oil Palm Tree. Retrieved from https://oilpalmblog.wordpress.com/2013/12/15/the-oil-palm-tree/

12. Akvopedia. (2018, 4 January). Sustainable Oil Palm Farming / Pruning. Retrieved from https://akvopedia.org/wiki/Sustainable_Oil_Palm_Farming_/_Pruning

13. Zakaria Azman. (2016, March 7). Electric E-Cutter can reach oil palm trees as high as 30 feet. Retrieved from https://phys.org/news/2016-03-electric-e-cutter-oil-palm-trees.html

14. Sowat, S. N., Ismail, W. I., Mahadi, M. R., Bejo, S. K., & Kassim, M. S. (2018). Trend In The Development Of Oil Palm Fruit Harvesting Technologies In Malaysia. *Jurnal Teknologi, 80*(2). doi:10.11113/jt.v80.11298

APPENDICES

TREE 1

Figure 7: Angle A Original image

Method 1 (Angle A)

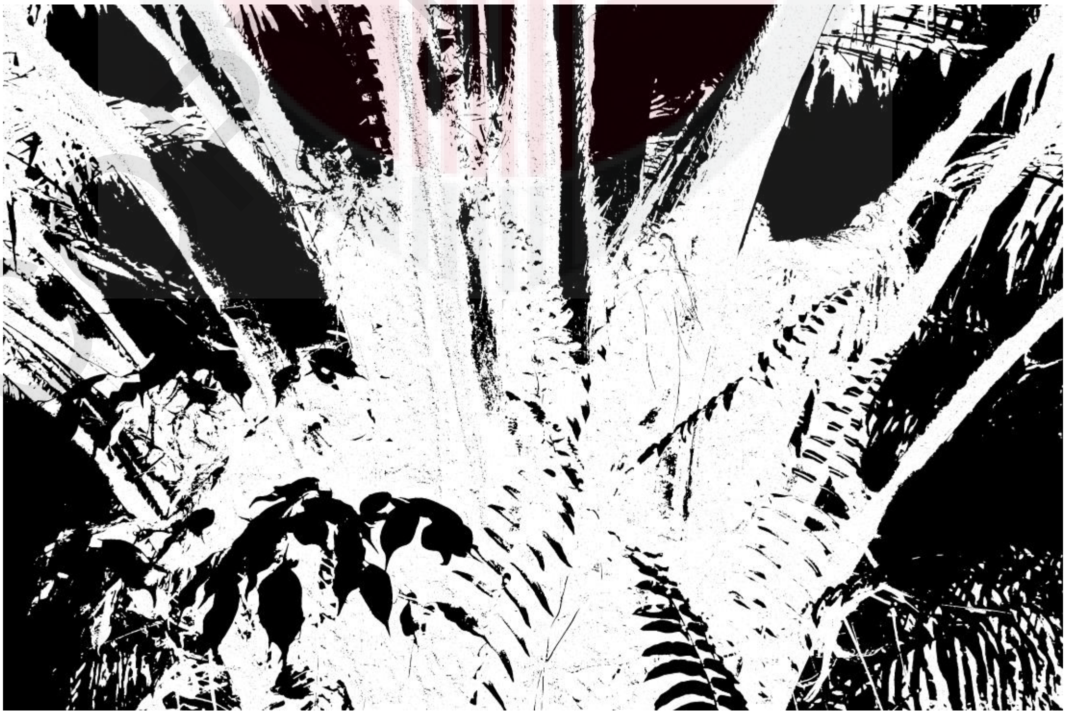


Figure 8: Final binarized masked image

Method 2 (Angle A)

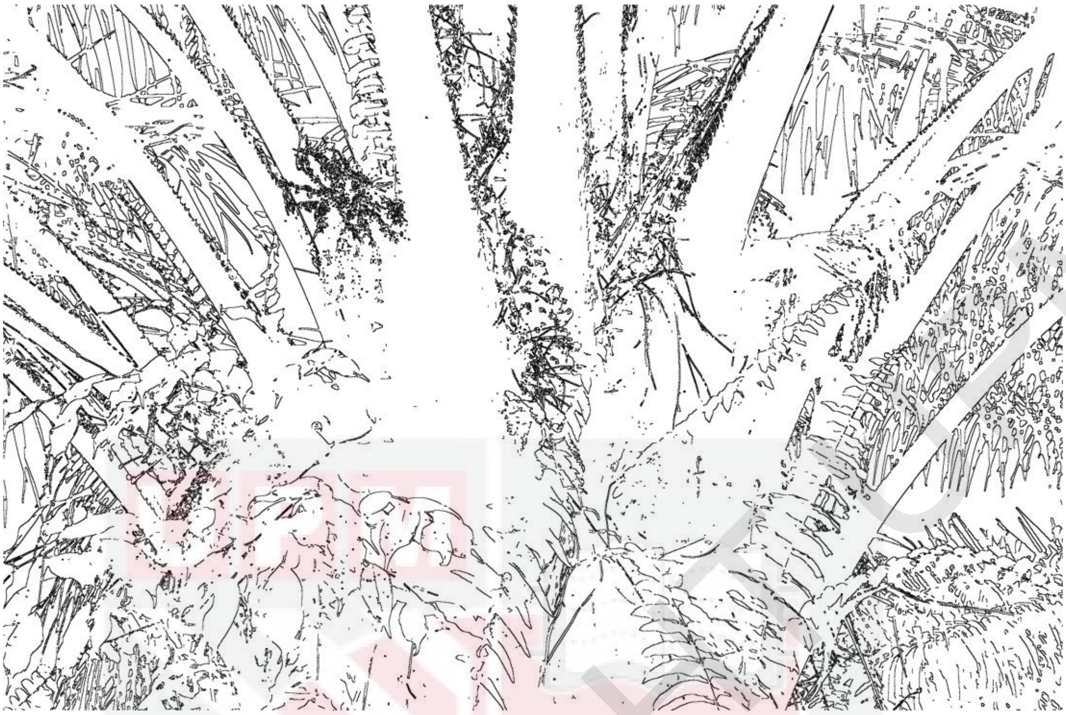


Figure 9: Final detected image

Method 3 (Angle A)

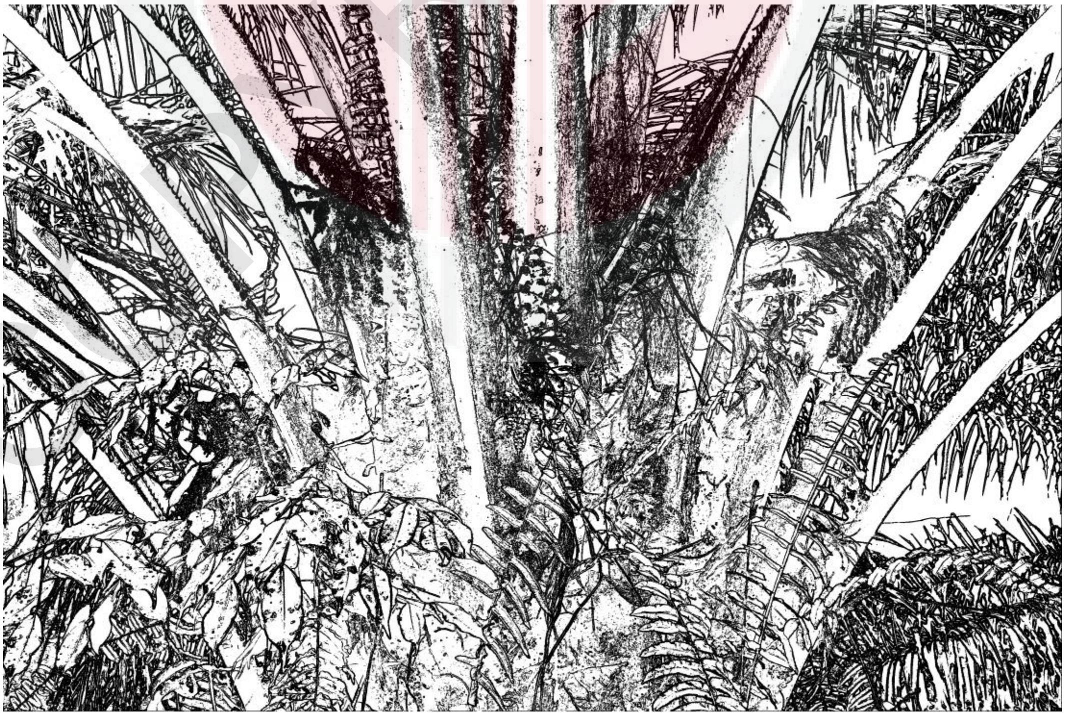


Figure 10: Final filtered image

Figure 11: Angle B Original image

Method 1 (Angle B)

Figure 12: Final binarized masked image

Method 2 (Angle B)

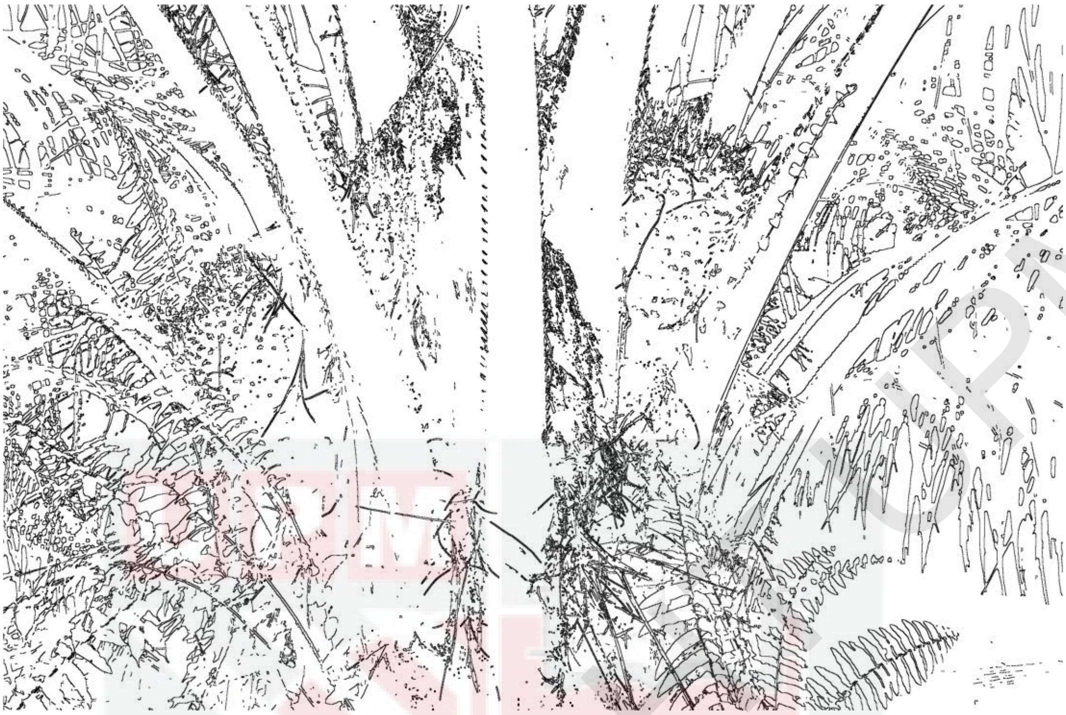


Figure 13: Final detected image

Method 3 (Angle B)

Figure 14: Final filtered image

Figure 15: Angle C Original image

Method 1 (Angle C)

Figure 16: Final binarized masked image

Method 2 (Angle C)

Figure 17: Final detected image

Method 3 (Angle C)

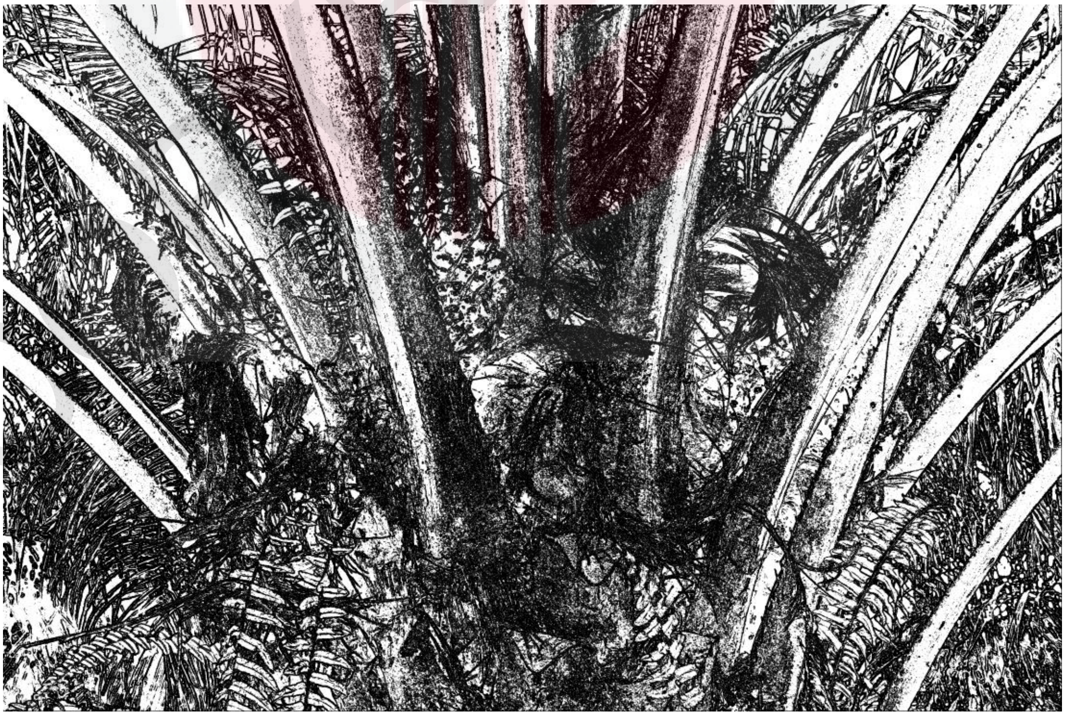


Figure 18: Final filtered image

Figure 19: Angle D Original image

Method 1 (Angle D)

Figure 20: Final binarized masked image

Method 2 (Angle D)

Figure 21: Final detected image

Method 3 (Angle D)

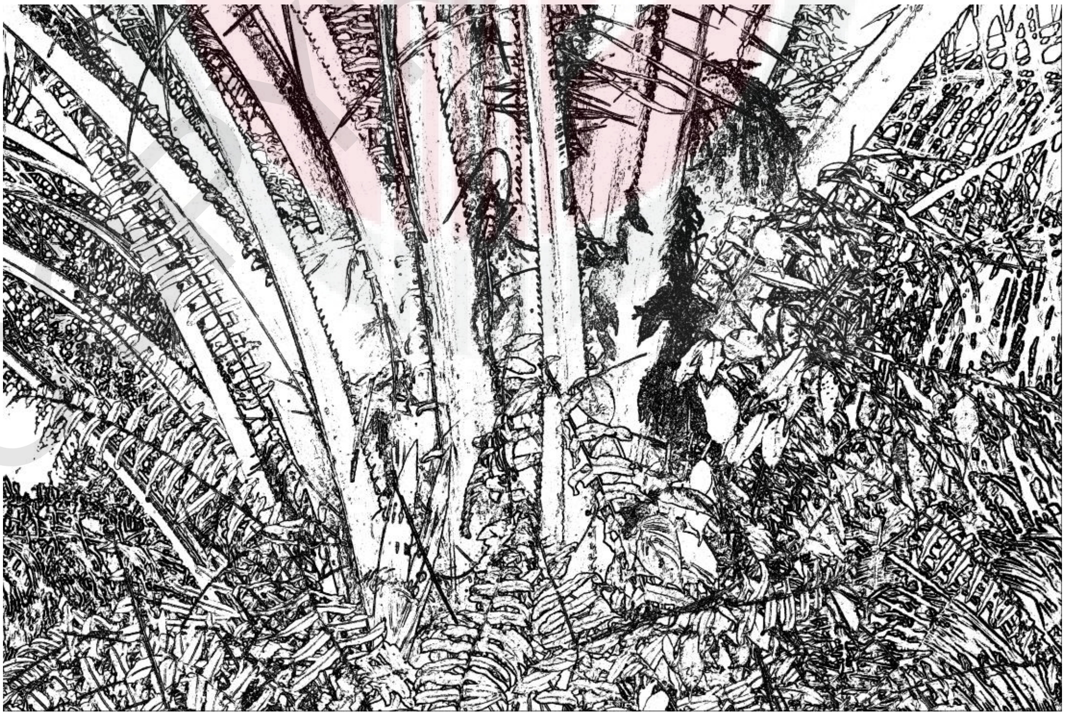


Figure 22: Final filtered image

TREE 2

Figure 23: Angle A Original image

Method 1 (Angle A)

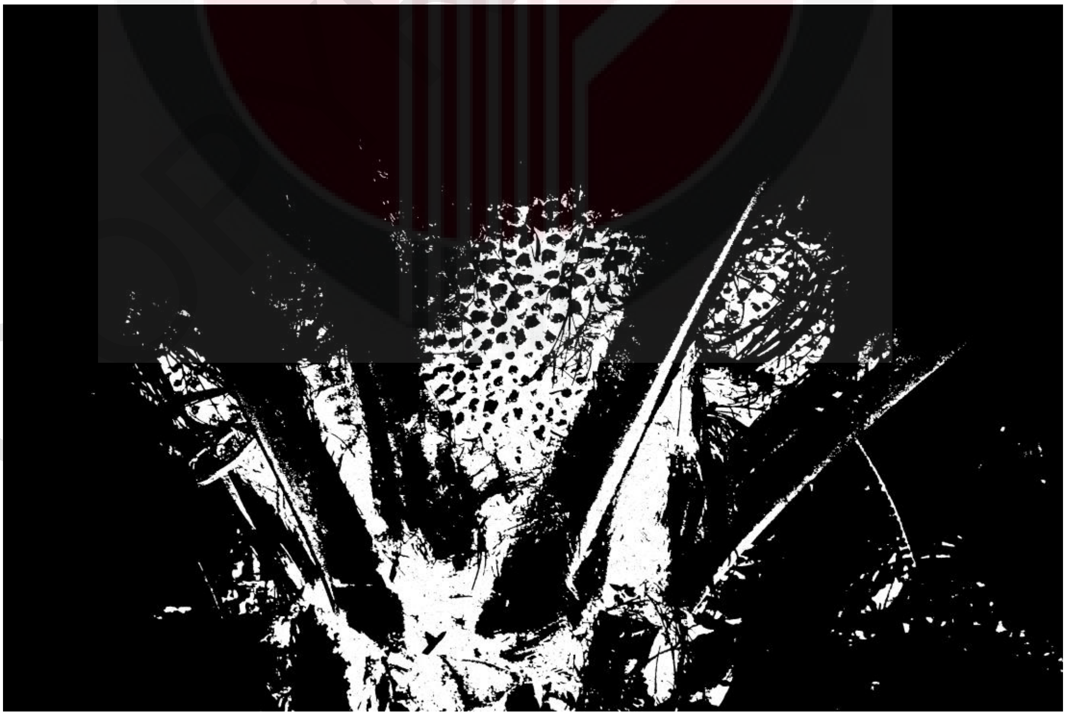


Figure 24: Final binarized masked image

Method 2 (Angle A)

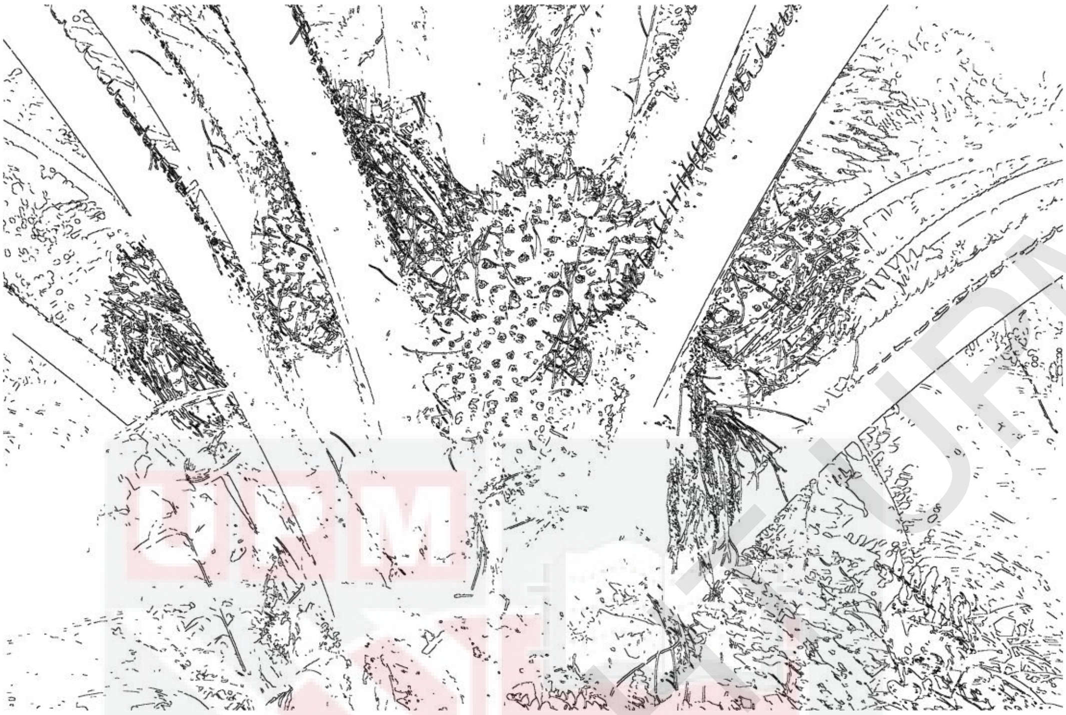


Figure 25: Final detected image

Method 3 (Angle A)

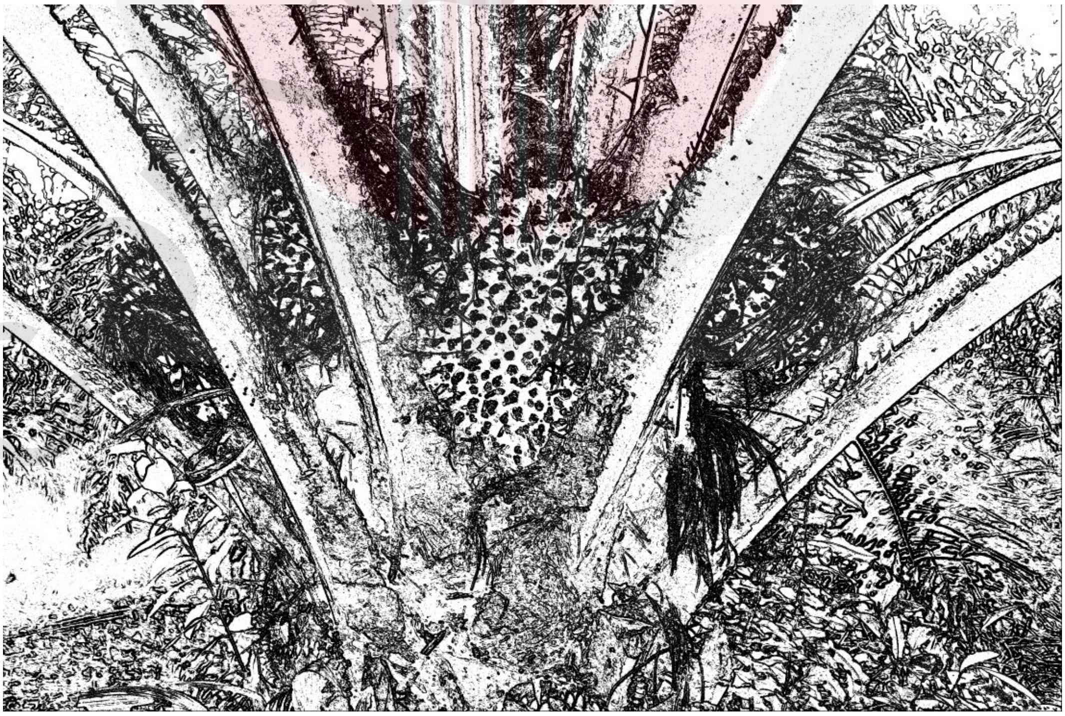


Figure 26: Final filtered image

Figure 27: Angle B Original image

Method 1 (Angle B)

Figure 28: Final binarized masked image

Method 2 (Angle B)

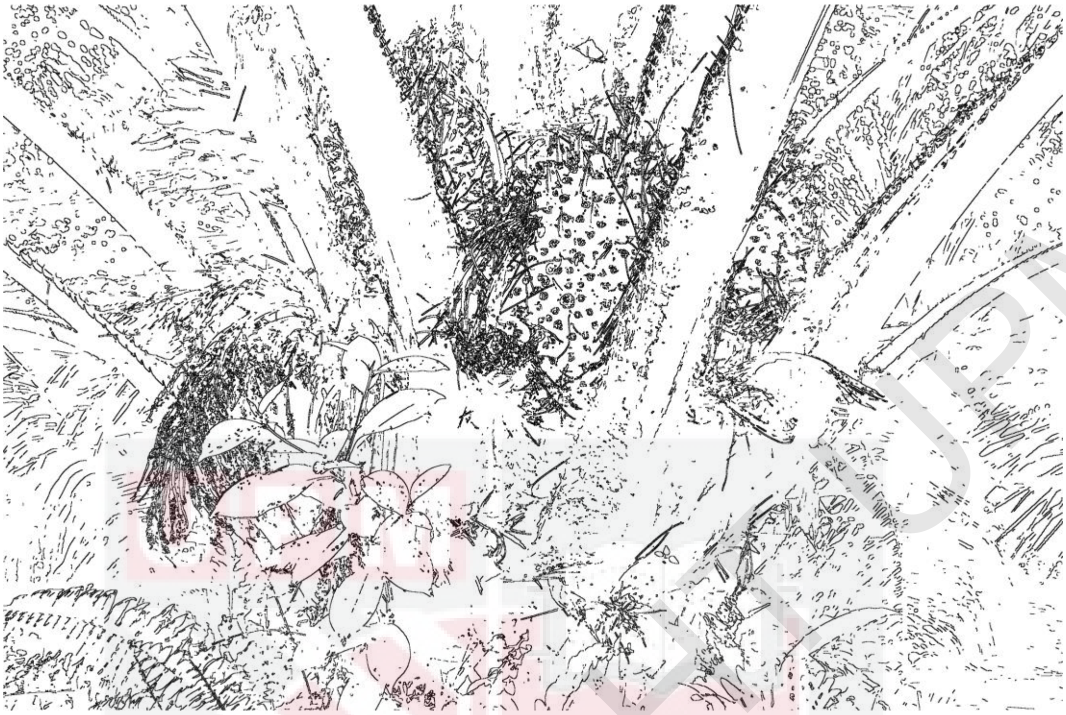


Figure 29: Final detected image

Method 3 (Angle B)

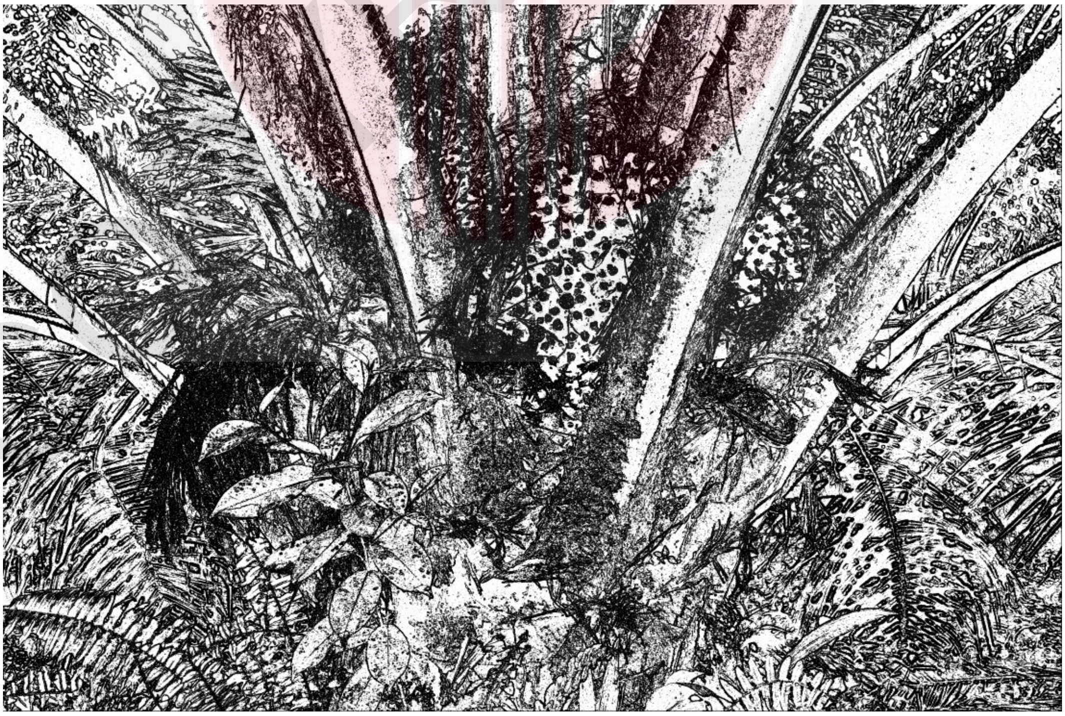


Figure 30: Final filtered image

Figure 31: Angle C Original image

Method 1 (Angle C)

Figure 32: Final binarized masked image

Method 2 (Angle C)

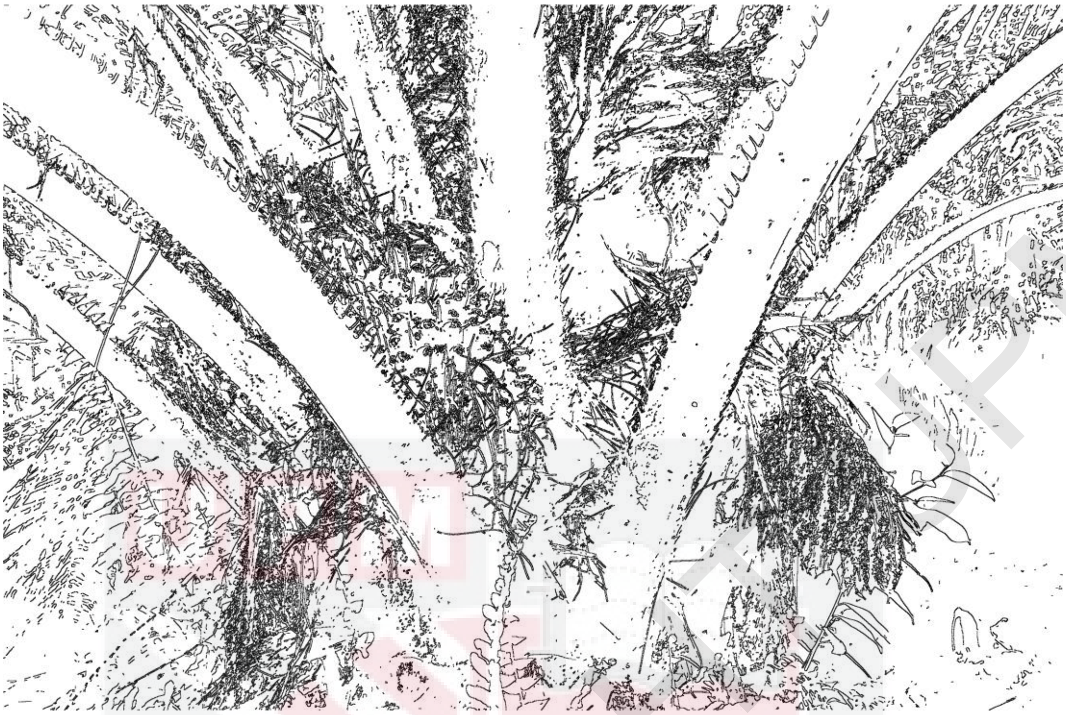


Figure 33: Final detected image

Method 3 (Angle C)

Figure 34: Final filtered image

Figure 35: Angle D Original image

Method 1 (Angle D)

Figure 36: Final binarized masked image

Method 2 (Angle D)

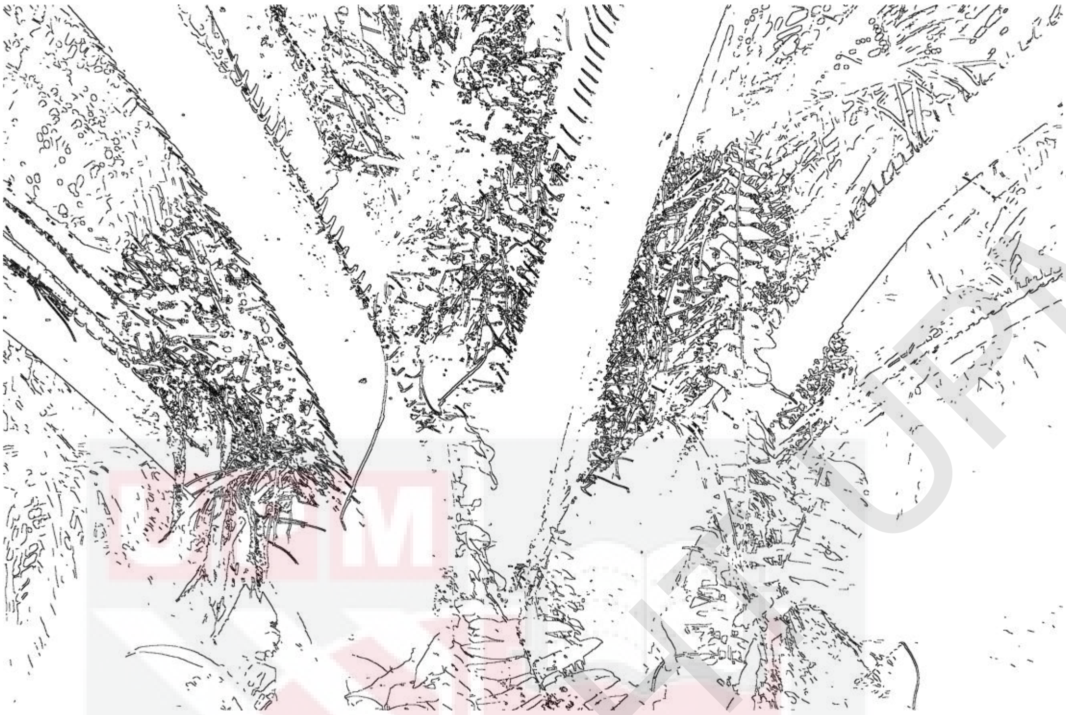


Figure 37: Final detected image

Method 3 (Angle D)

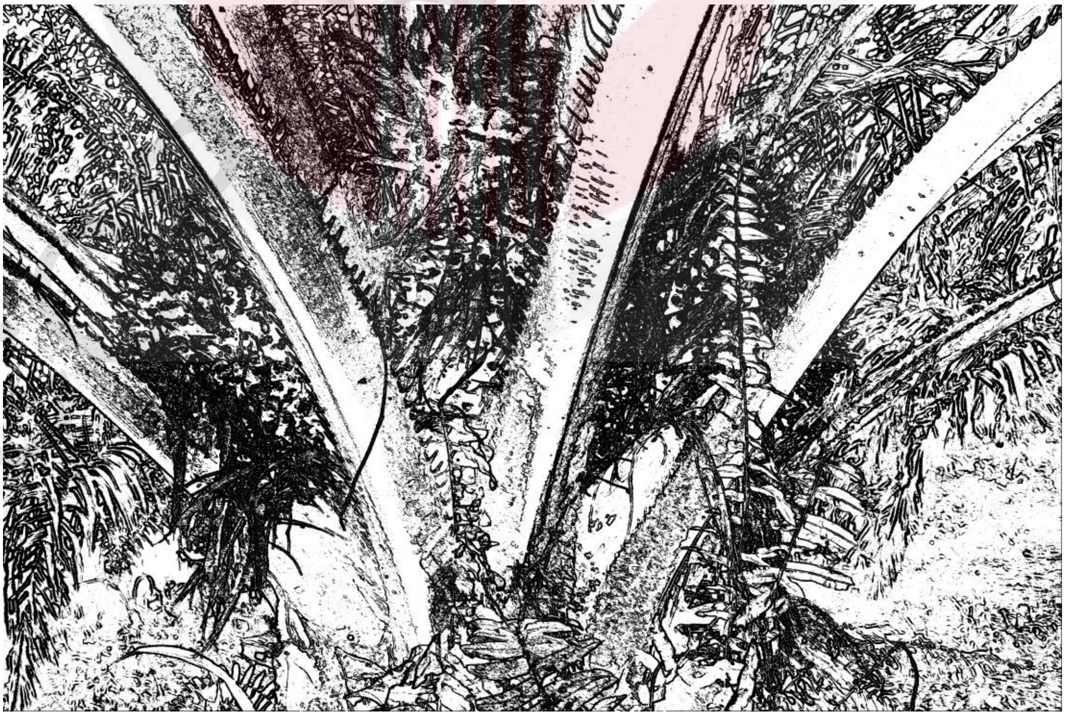


Figure 38: Final filtered image

TREE 3

Figure 39: Angle A Original image

Method 1 (Angle A)

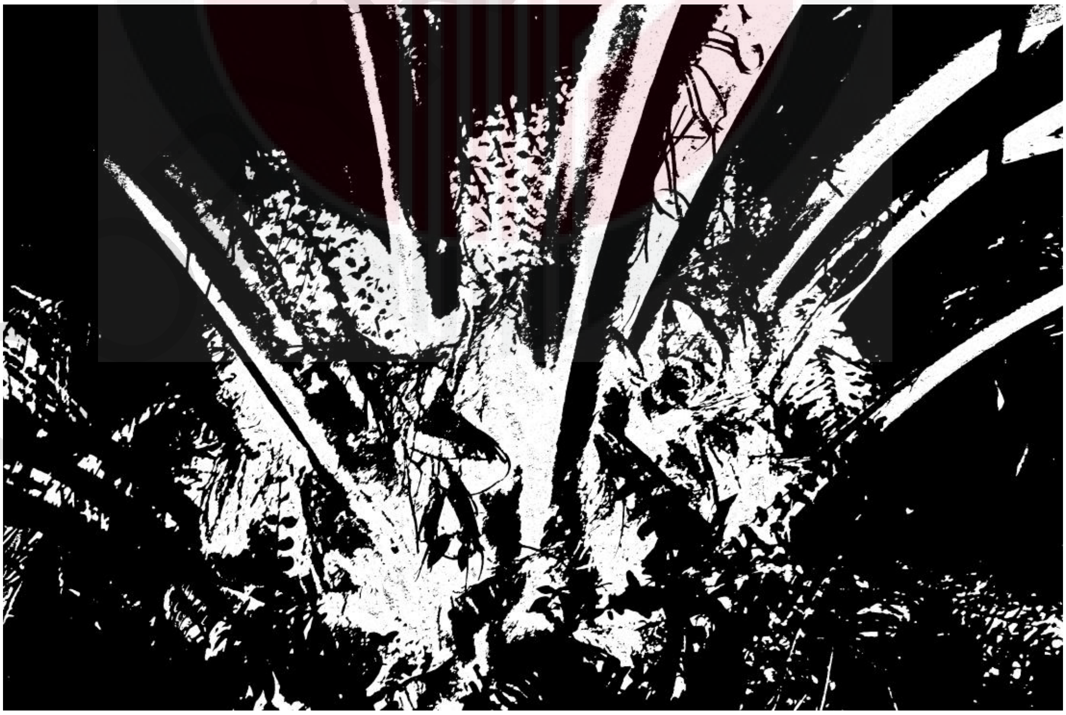


Figure 40: Final binarized masked image

Method 2 (Angle A)

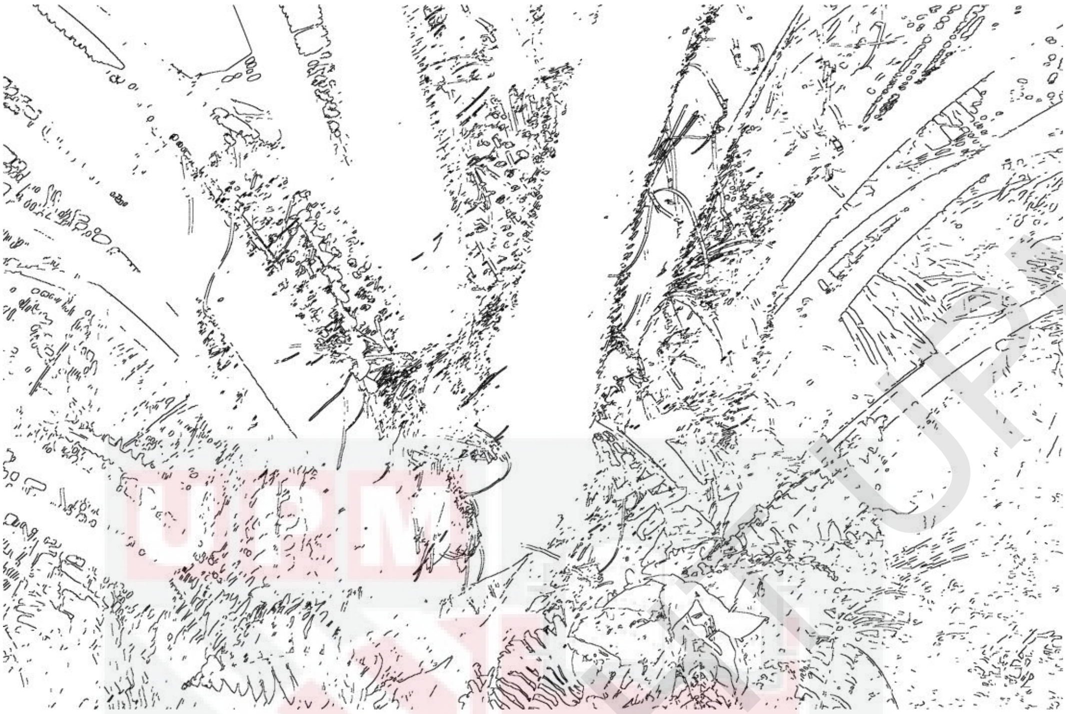


Figure 41: Final detected image

Method 3 (Angle A)

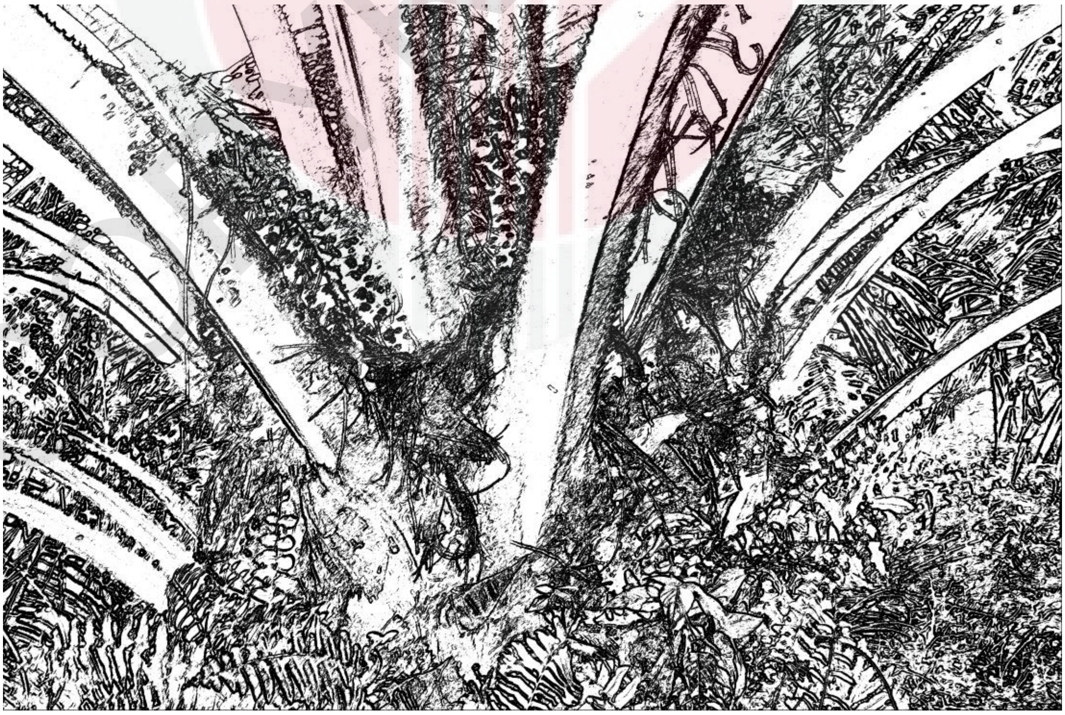


Figure 42: Final filtered image

Figure 43: Angle B Original image

Method 1 (Angle B)

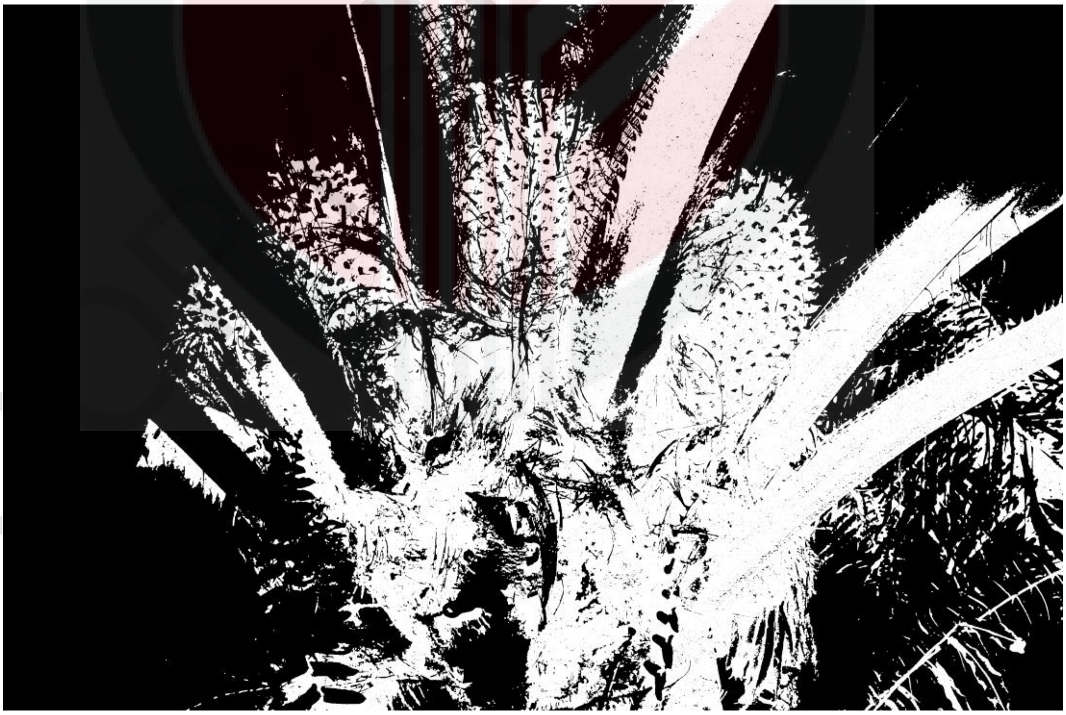


Figure 44: Final binarized masked image

Method 2 (Angle B)

Figure 45: Final detected image

Method 3 (Angle B)

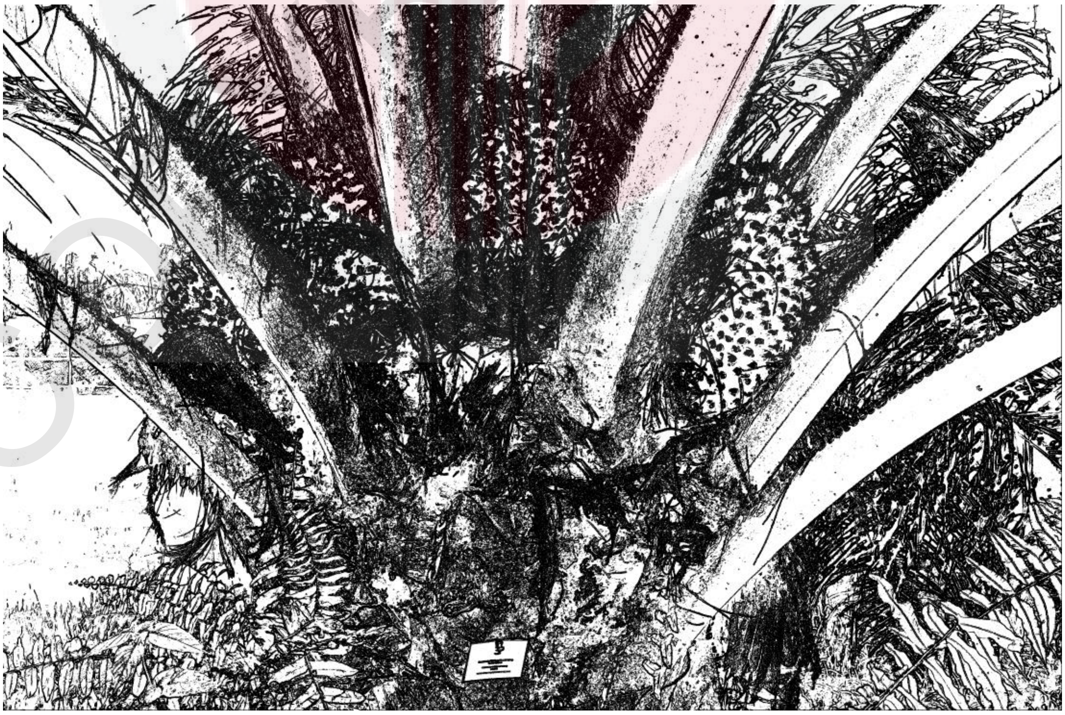


Figure 46: Final filtered image

Figure 47: Angle C Original image

Method 1 (Angle C)

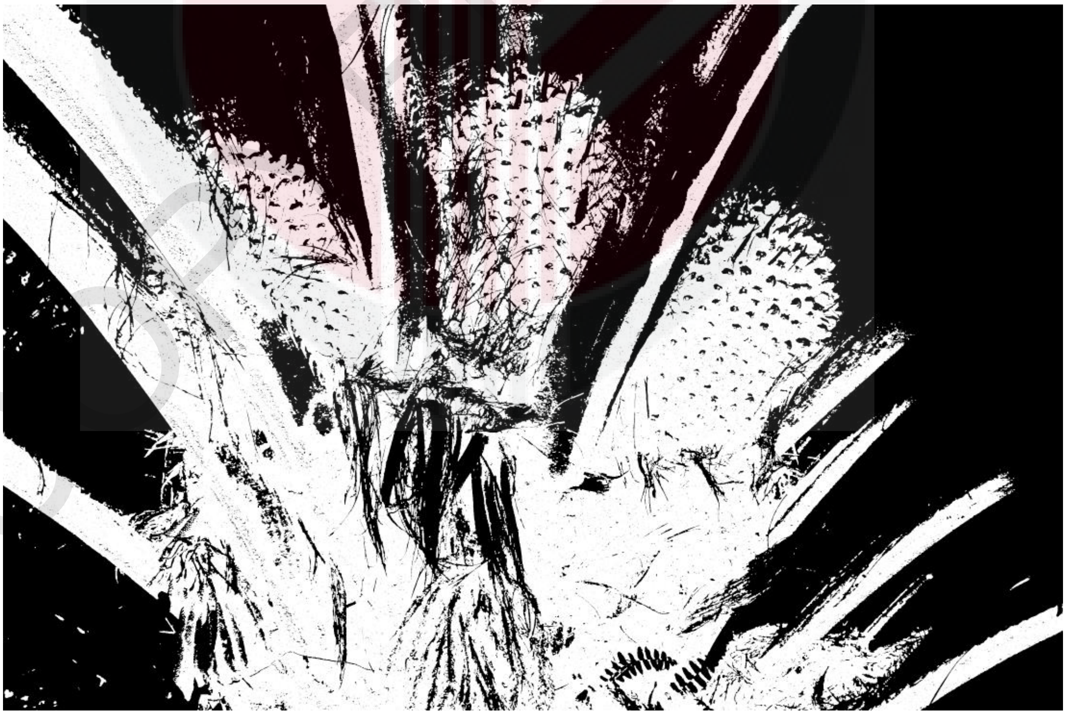


Figure 48: Final binarized masked image

Method 2 (Angle C)

Figure 49: Final detected image

Method 3 (Angle C)

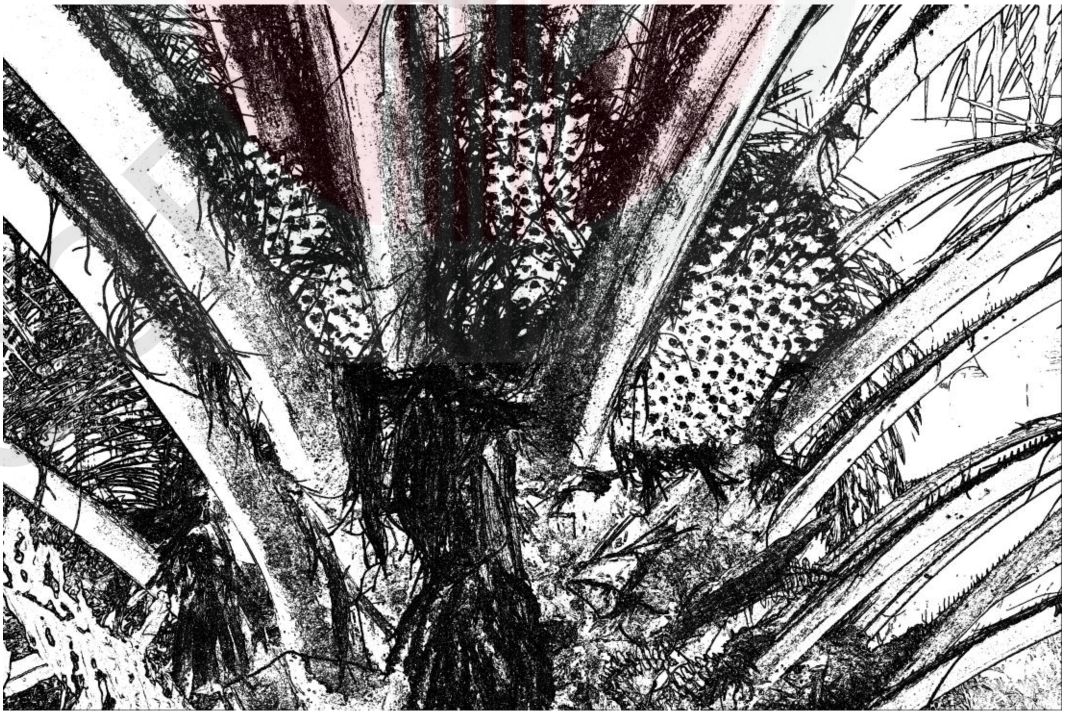


Figure 50: Final filtered image

Figure 51: Angle D Original image

Method 1 (Angle D)

Figure 52: Final binarized masked image

Method 2 (Angle D)

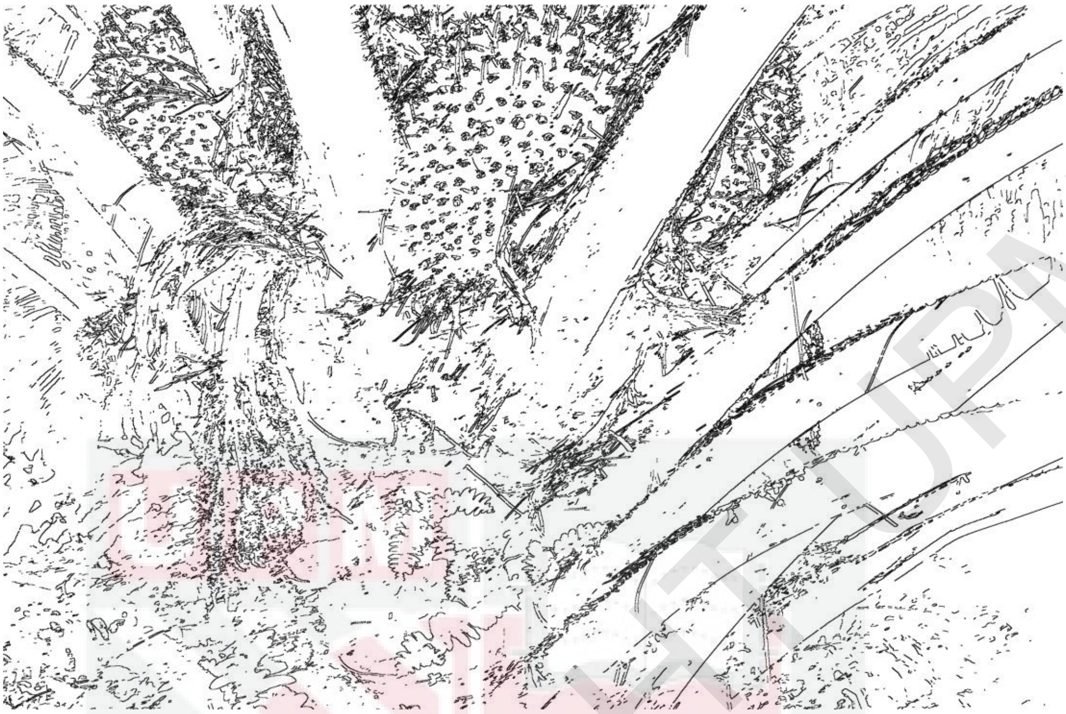


Figure 53: Final detected image

Method 3 (Angle D)

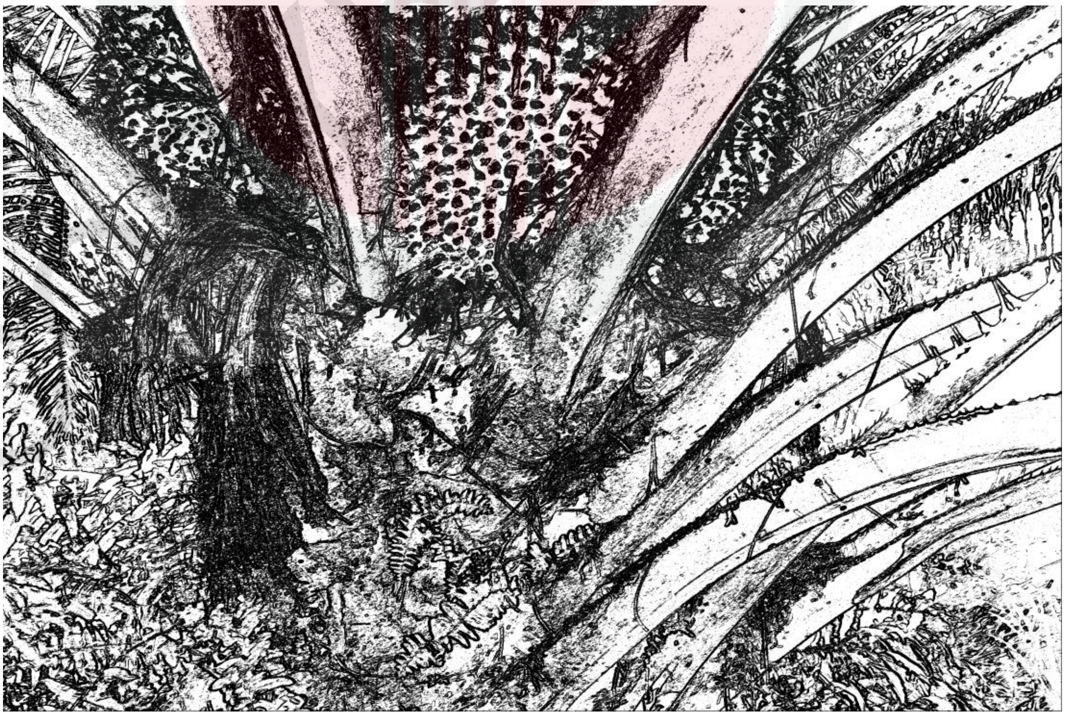


Figure 54: Final filtered image

TREE 4

Figure 55: Angle A Original image

Method 1 (Angle A)

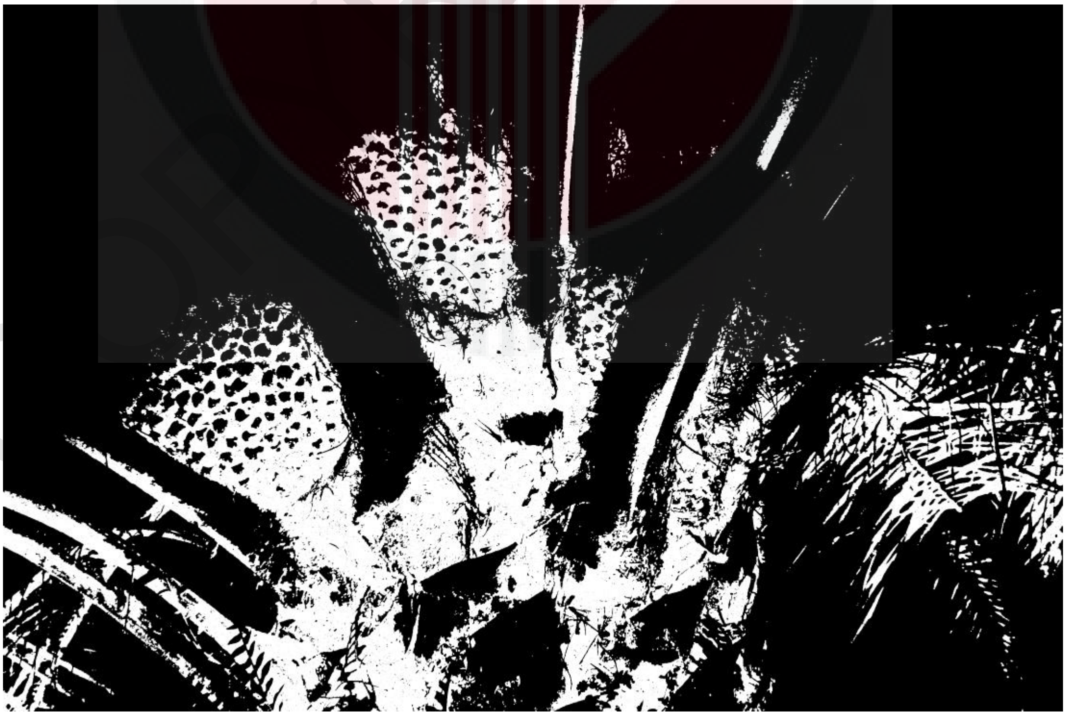


Figure 56: Final binarized masked image

Method 2 (Angle A)

Figure 57: Final detected image

Method 3 (Angle A)

Figure 58: Final filtered image

Figure 59: Angle B Original image

Method 1 (Angle B)

Figure 60: Final binarized masked image

Method 2 (Angle B)

Figure 61: Final detected image

Method 3 (Angle B)

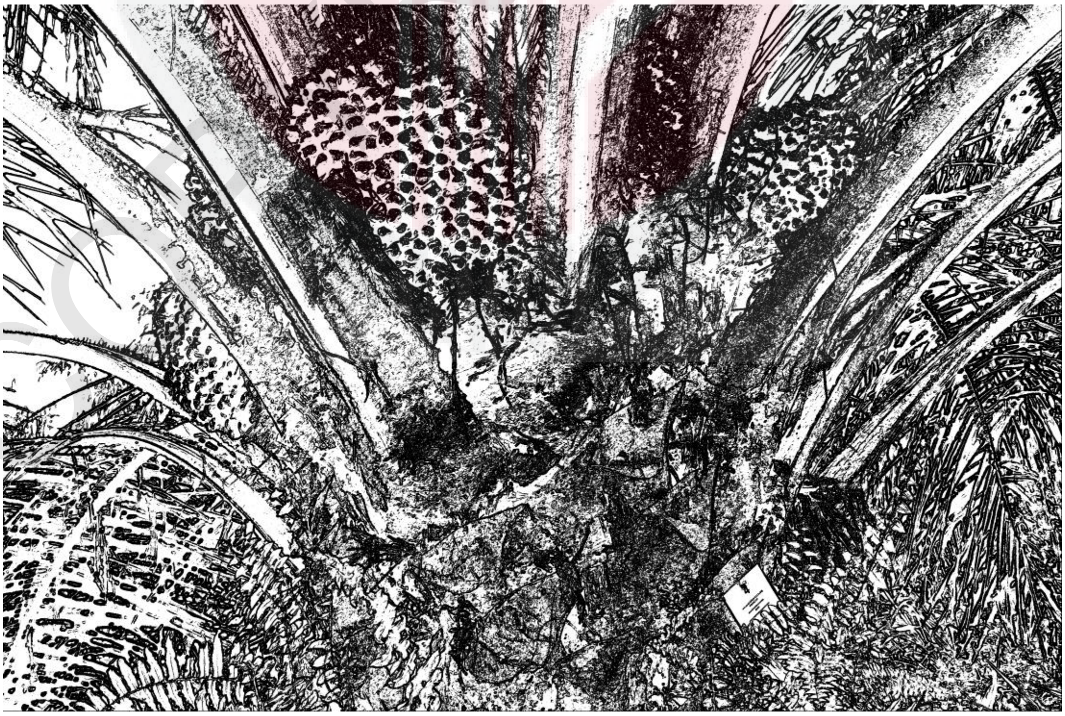


Figure 62: Final filtered image

Figure 63: Angle C Original image

Method 1 (Angle C)

Figure 64: Final binarized masked image

Method 2 (Angle C)

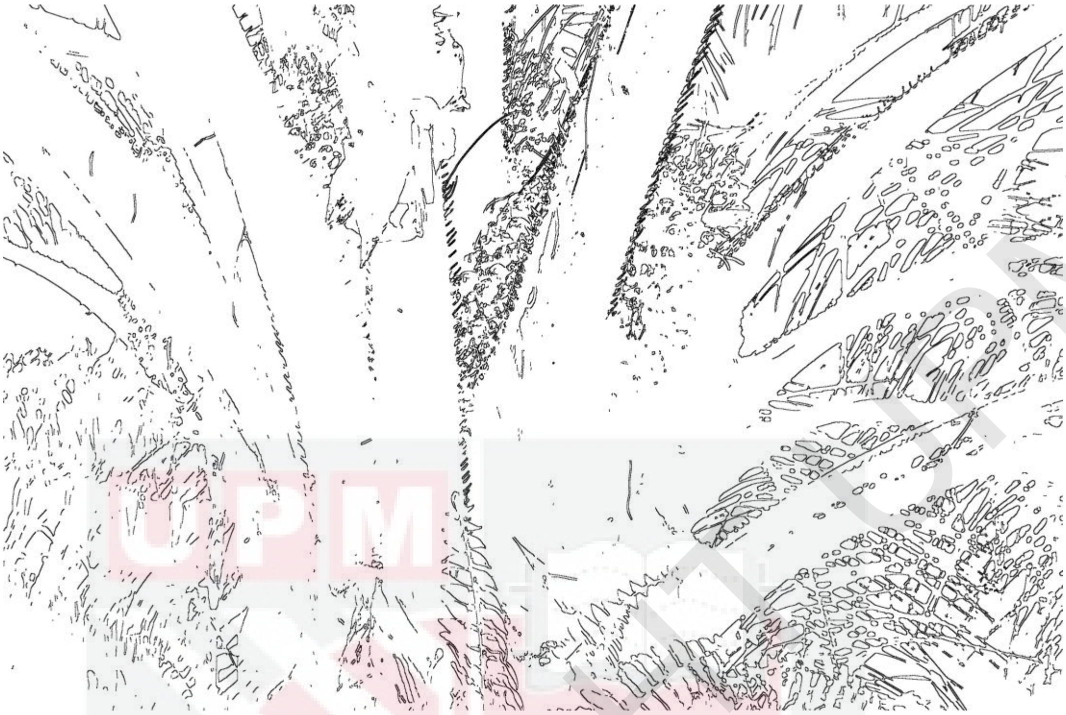


Figure 65: Final detected image

Method 3 (Angle C)

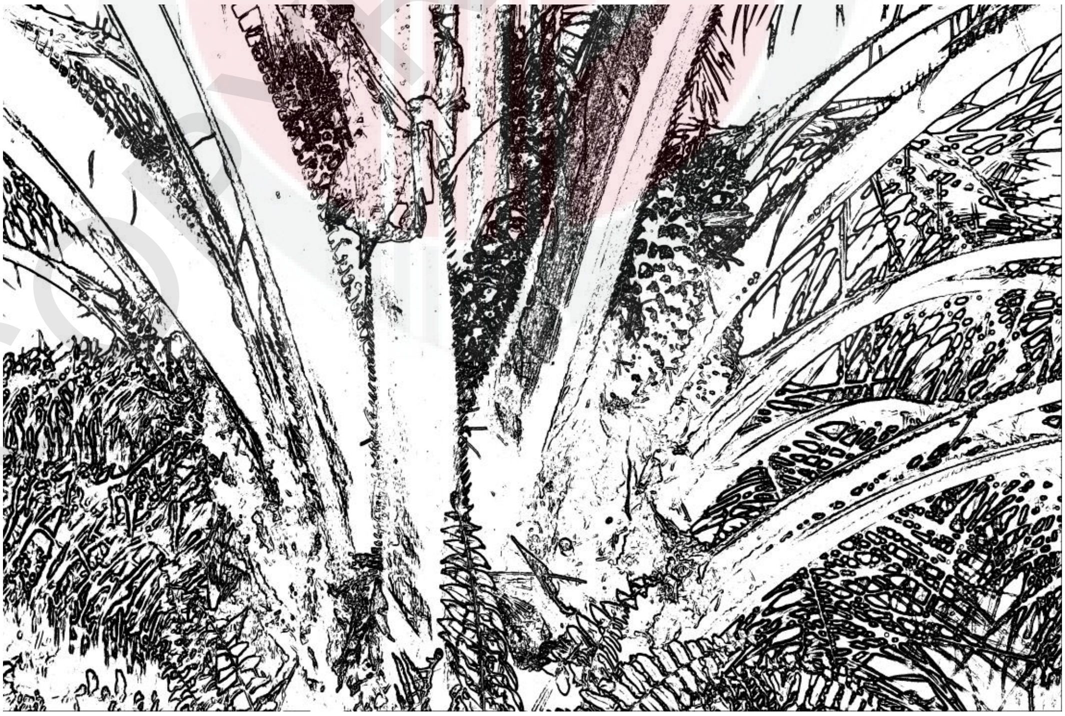


Figure 66: Final filtered image

Figure 67: Angle D Original image

Method 1 (Angle D)

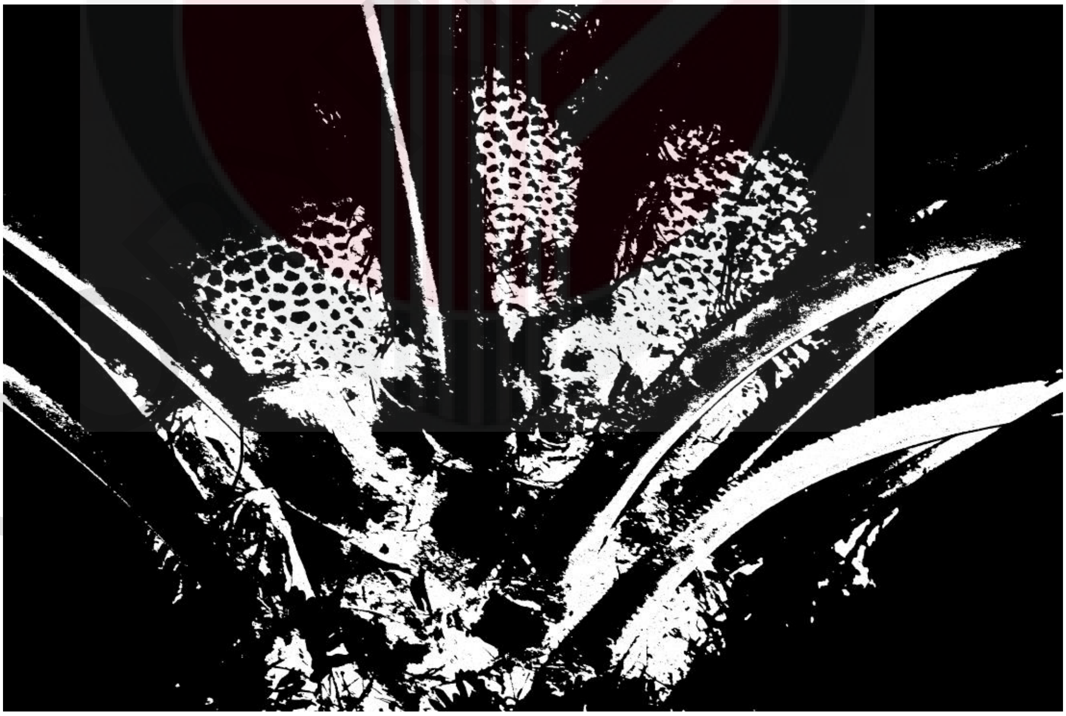


Figure 68: Final binarized masked image

Method 2 (Angle D)

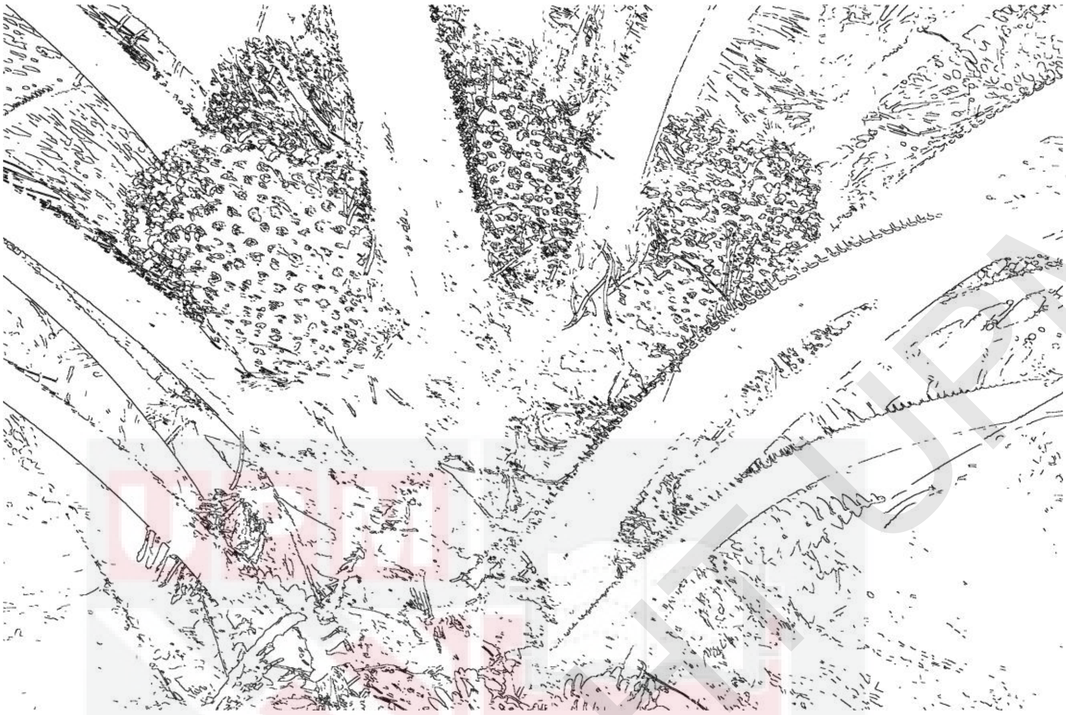


Figure 69: Final detected image

Method 3 (Angle D)

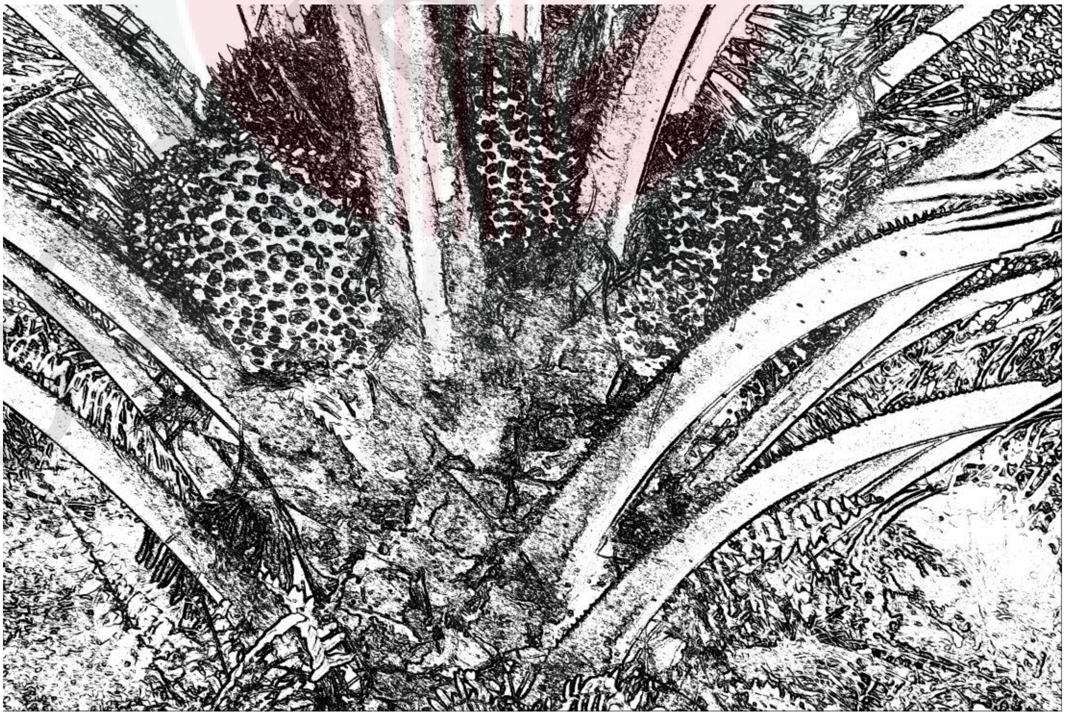


Figure 70: Final filtered image

TREE 5

Figure 71: Angle A Original image

Method 1 (Angle A)

Figure 72: Final binarized masked image

Method 2 (Angle A)

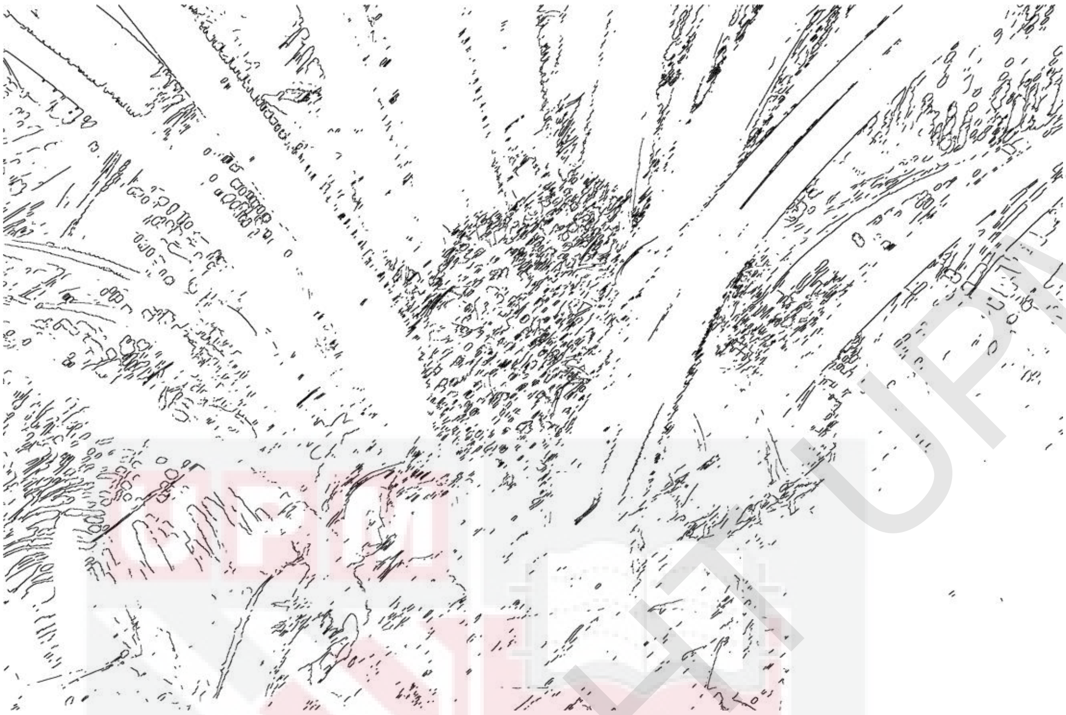


Figure 73: Final detected image

Method 3 (Angle A)

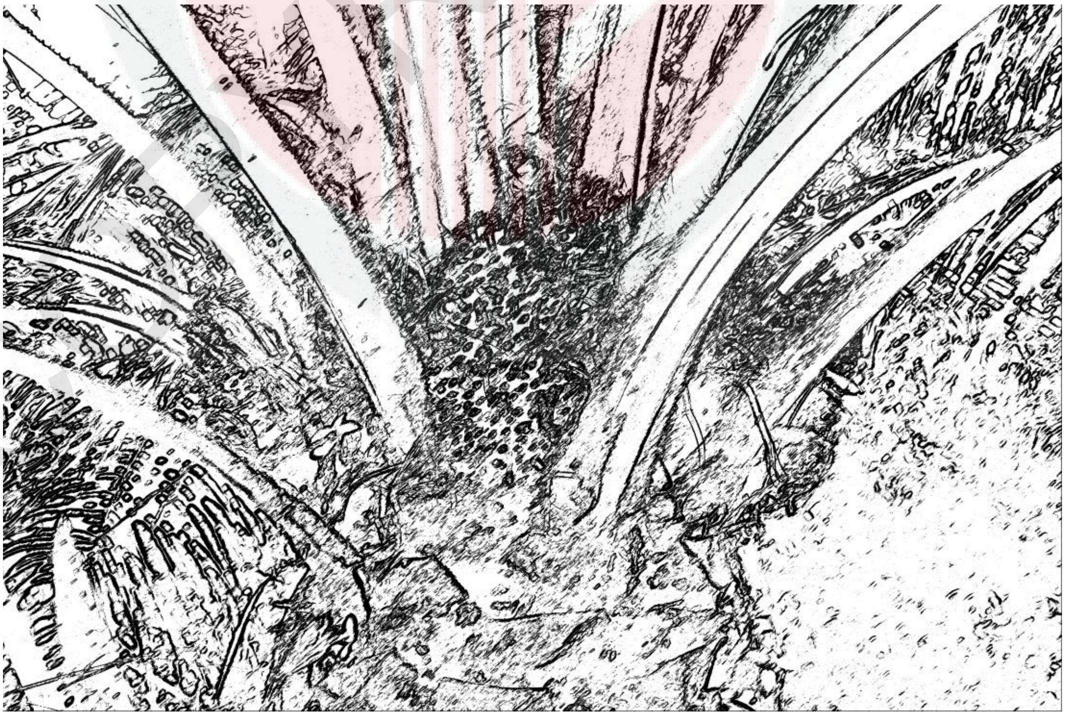


Figure 74: Final filtered image

Figure 75: Angle B Original image

Method 1 (Angle B)

Figure 76: Final binarized masked image

Method 2 (Angle B)

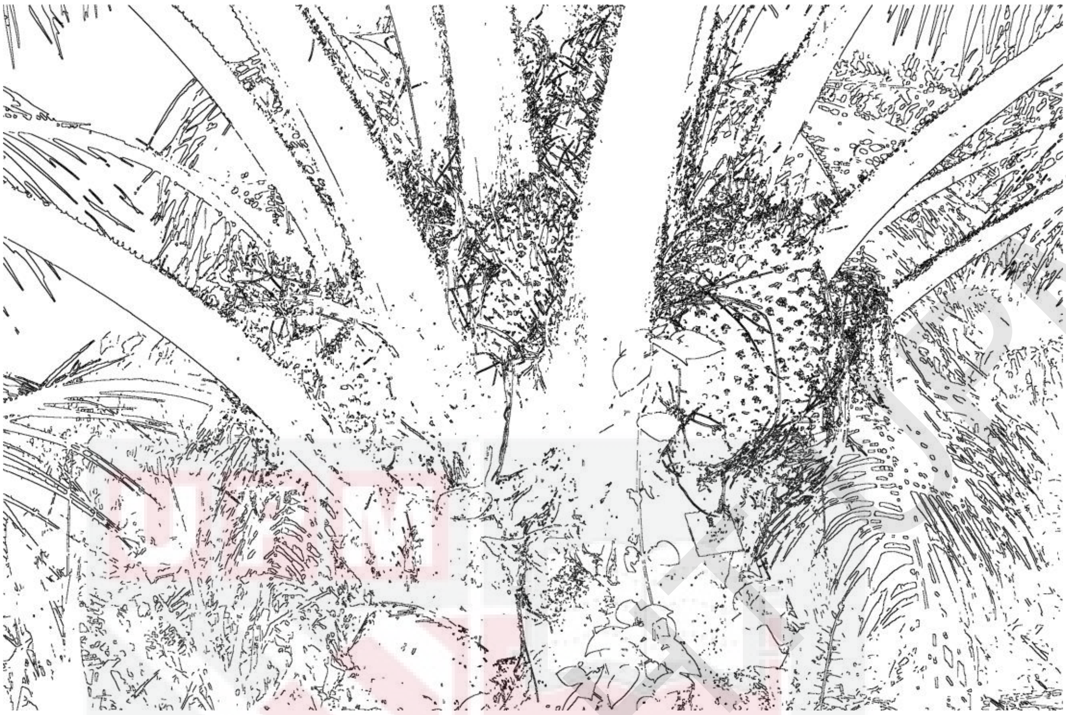


Figure 77: Final detected image

Method 3 (Angle B)

Figure 78: Final filtered image

Figure 79: Angle C Original image

Method 1 (Angle C)

Figure 80: Final binarized masked image

Method 2 (Angle C)

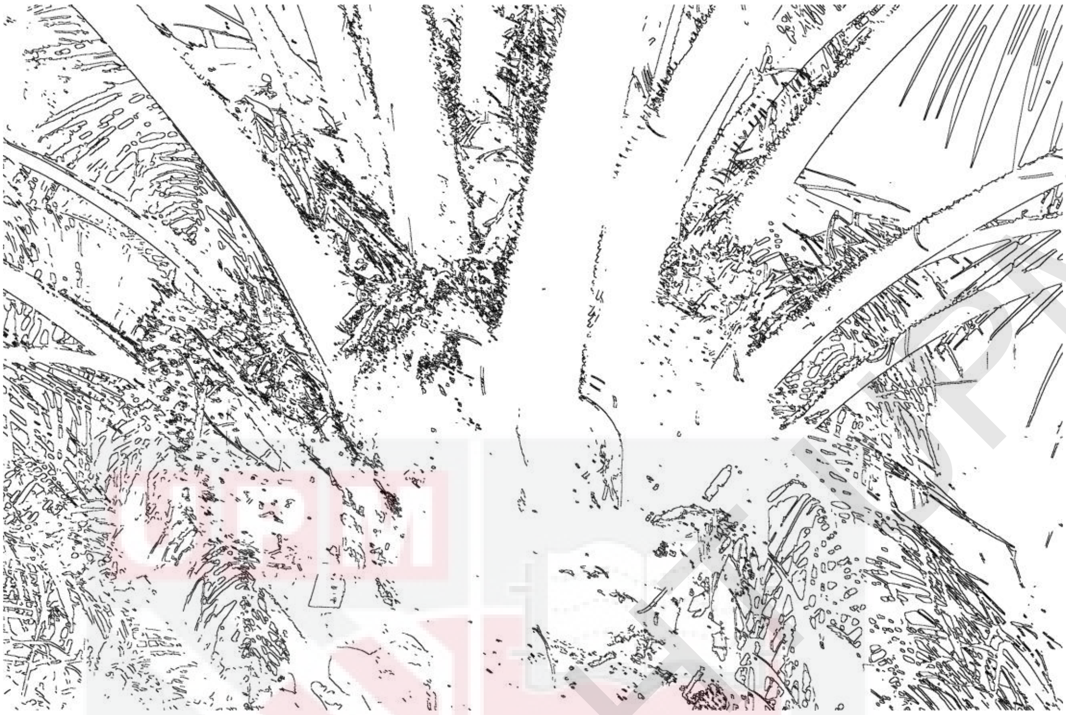


Figure 81: Final detected image

Method 3 (Angle C)

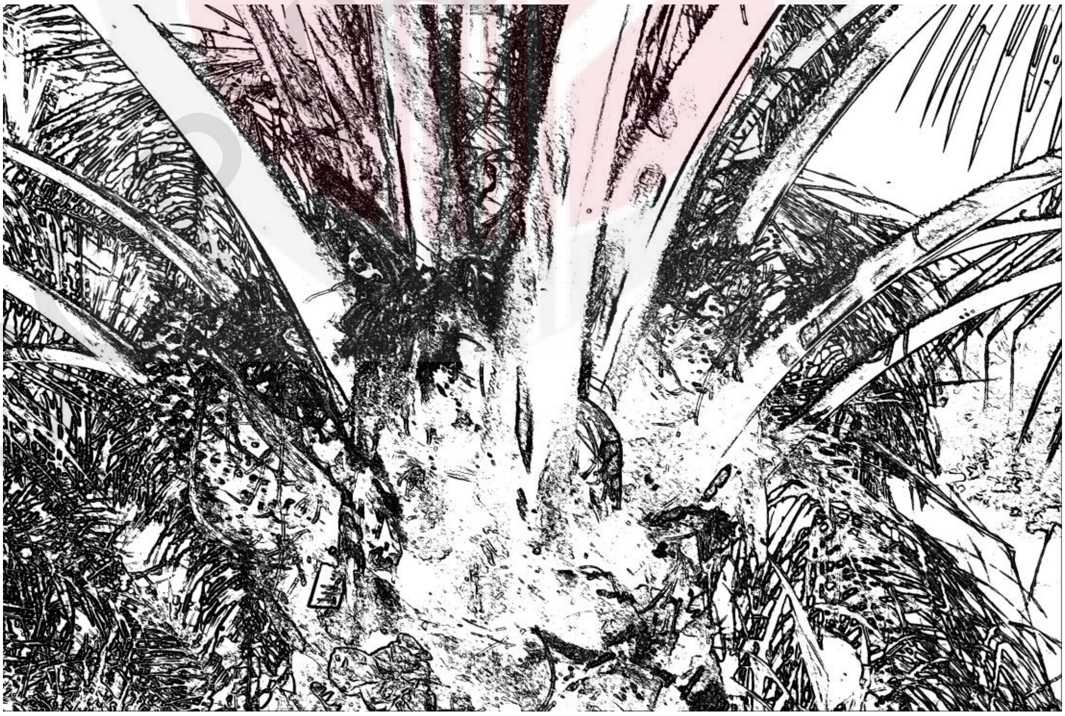


Figure 82: Final filtered image

Figure 83: Angle D Original image

Method 1 (Angle D)

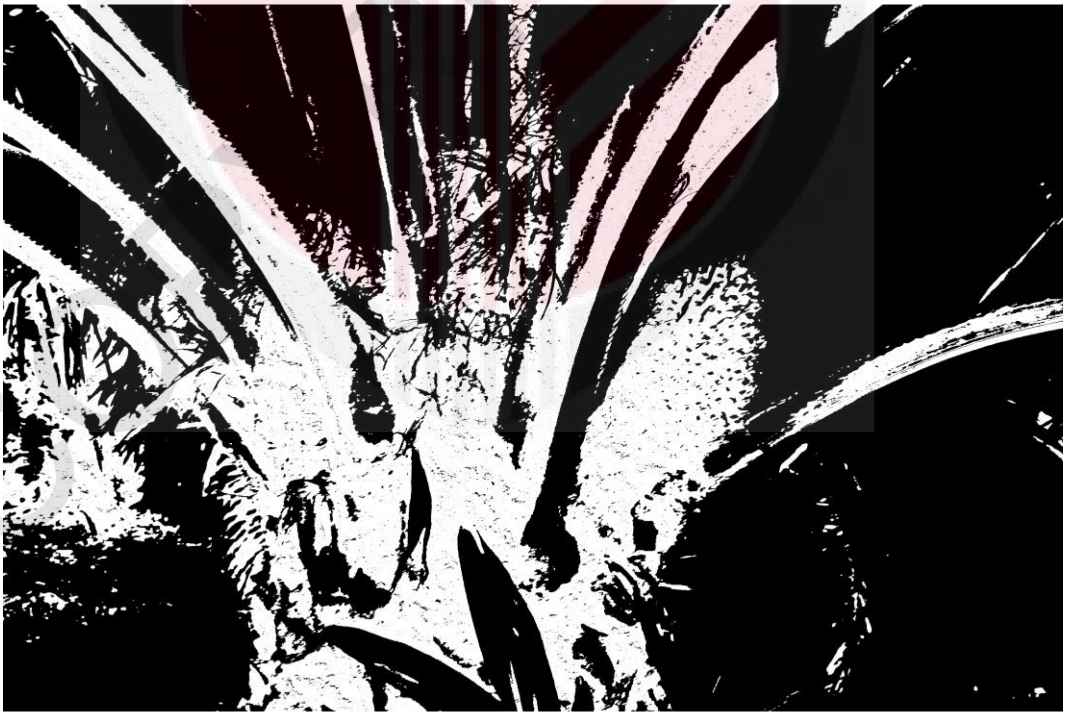


Figure 84: Final binarized masked image

Method 2 (Angle D)

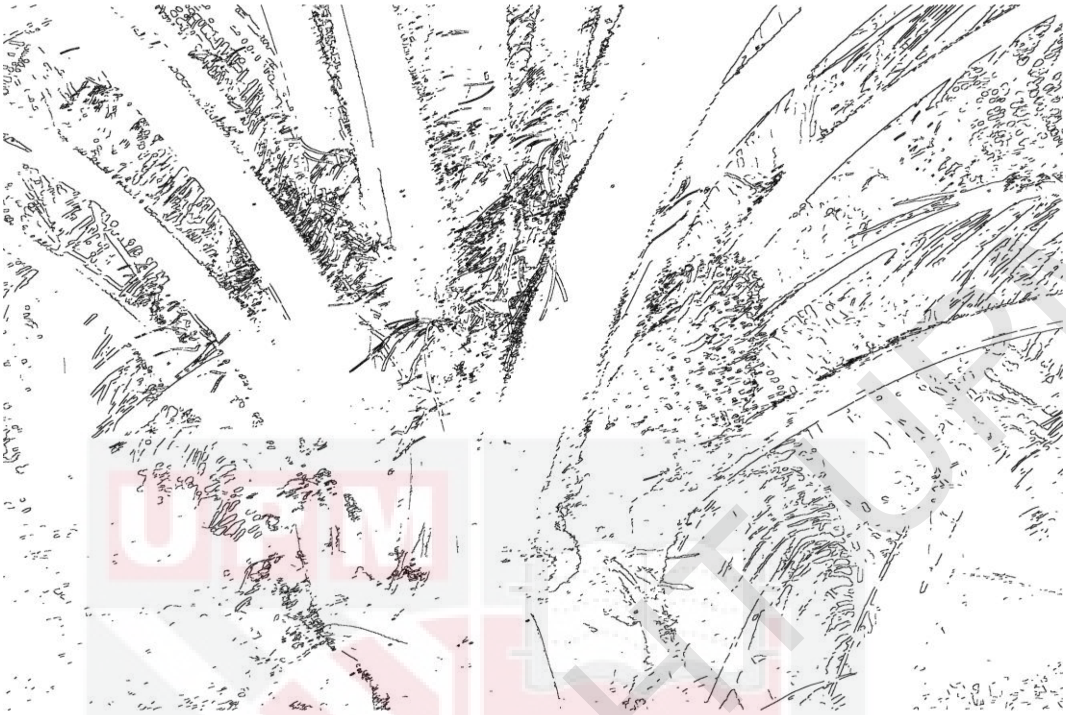


Figure 85: Final detected image

Method 3 (Angle D)

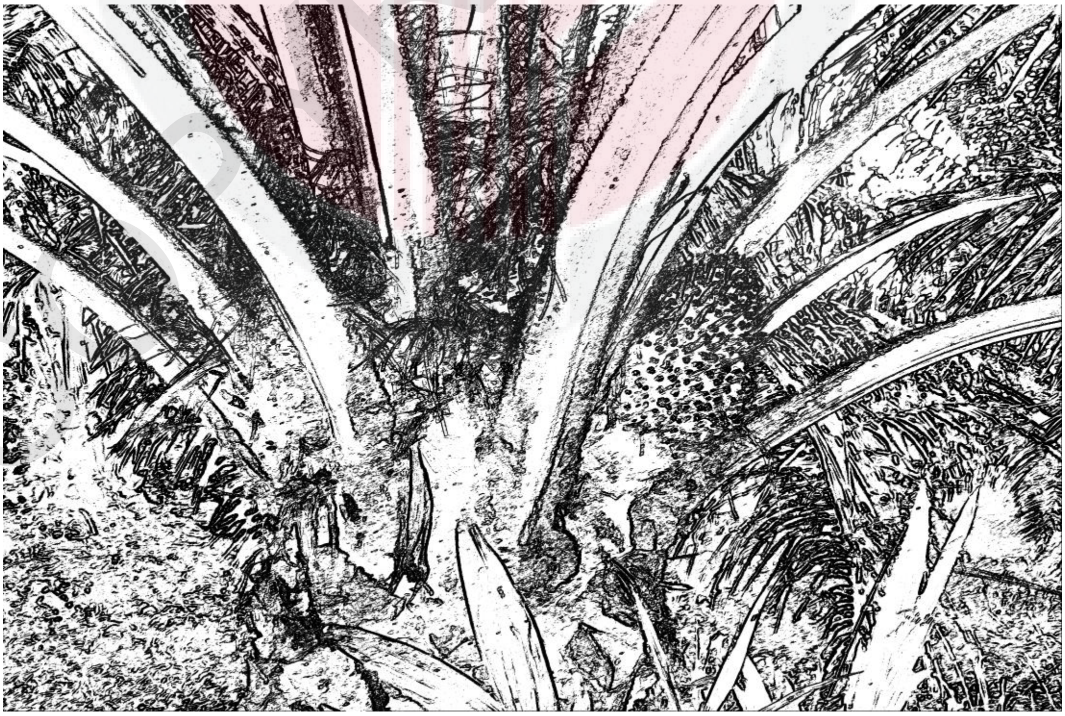


Figure 86: Final filtered image

TREE 6

Figure 87: Angle A Original image

Method 1 (Angle A)

Figure 88: Final binarized masked image

Method 2 (Angle A)

Figure 89: Final detected image

Method 3 (Angle A)

Figure 90: Final filtered image

Figure 91: Angle B Original image

Method 1 (Angle B)

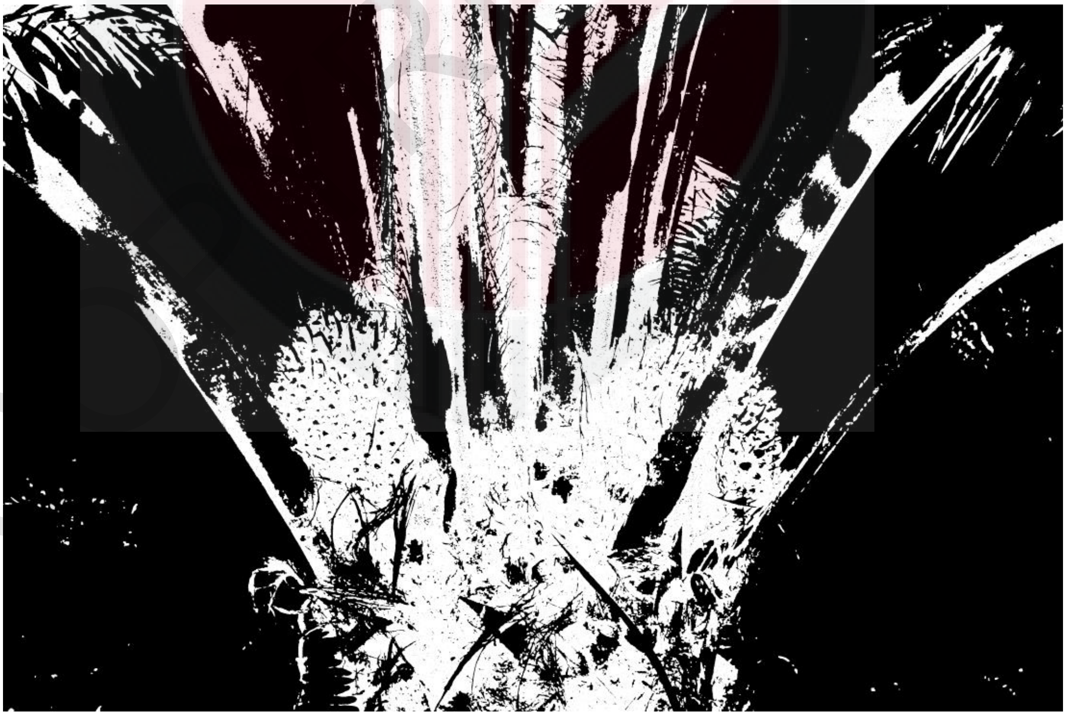


Figure 92: Final binarized masked image

Method 2 (Angle B)

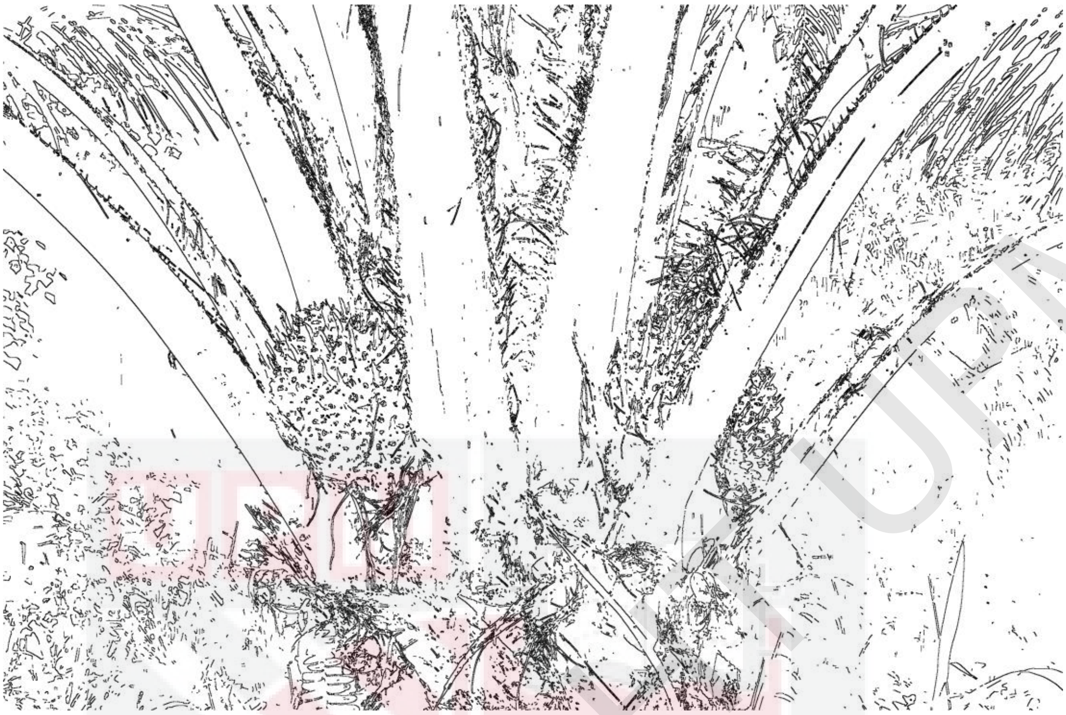


Figure 93: Final detected image

Method 3 (Angle B)

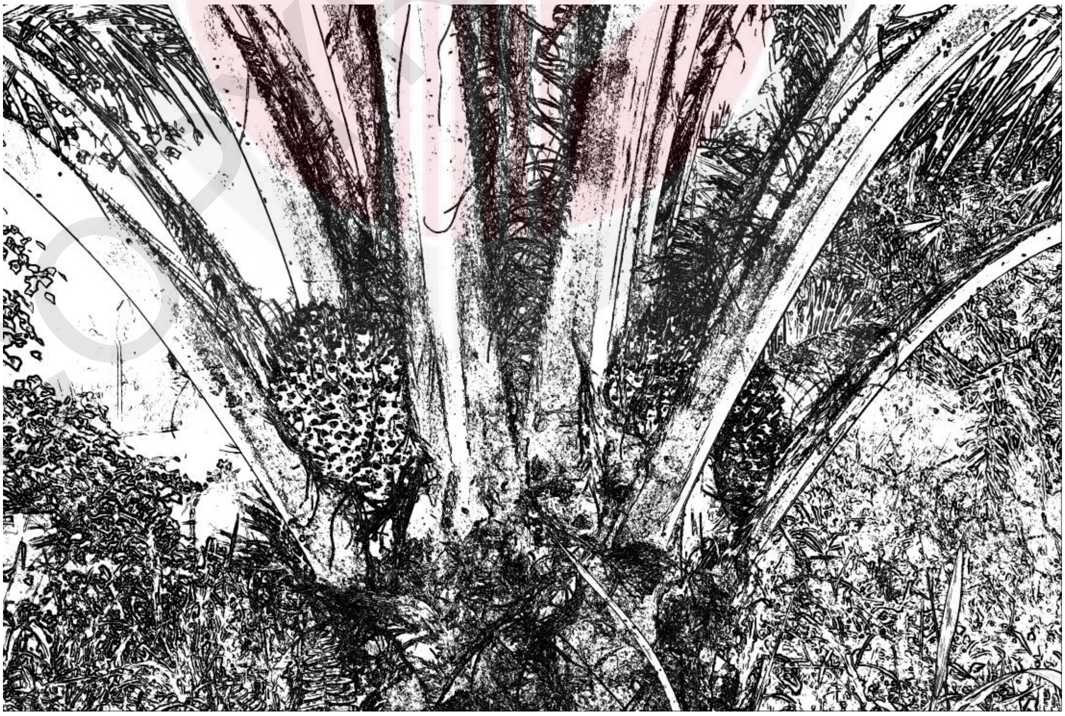


Figure 94: Final filtered image

Figure 95: Angle C Original image

Method 1 (Angle C)

Figure 96: Final binarized masked image

Method 2 (Angle C)

Figure 97: Final detected image

Method 3 (Angle C)

Figure 98: Final filtered image

Figure 99: Angle D Original image

Method 1 (Angle D)

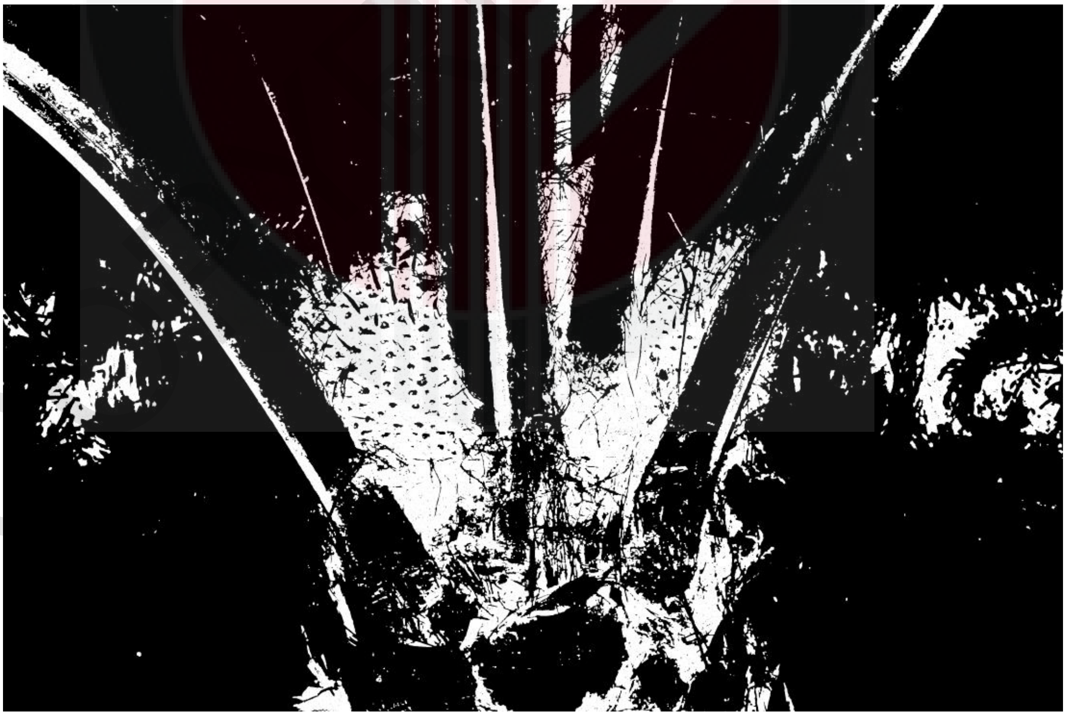


Figure 100: Final binarized masked image

Method 2 (Angle D)

Figure 101: Final detected image

Method 3 (Angle D)

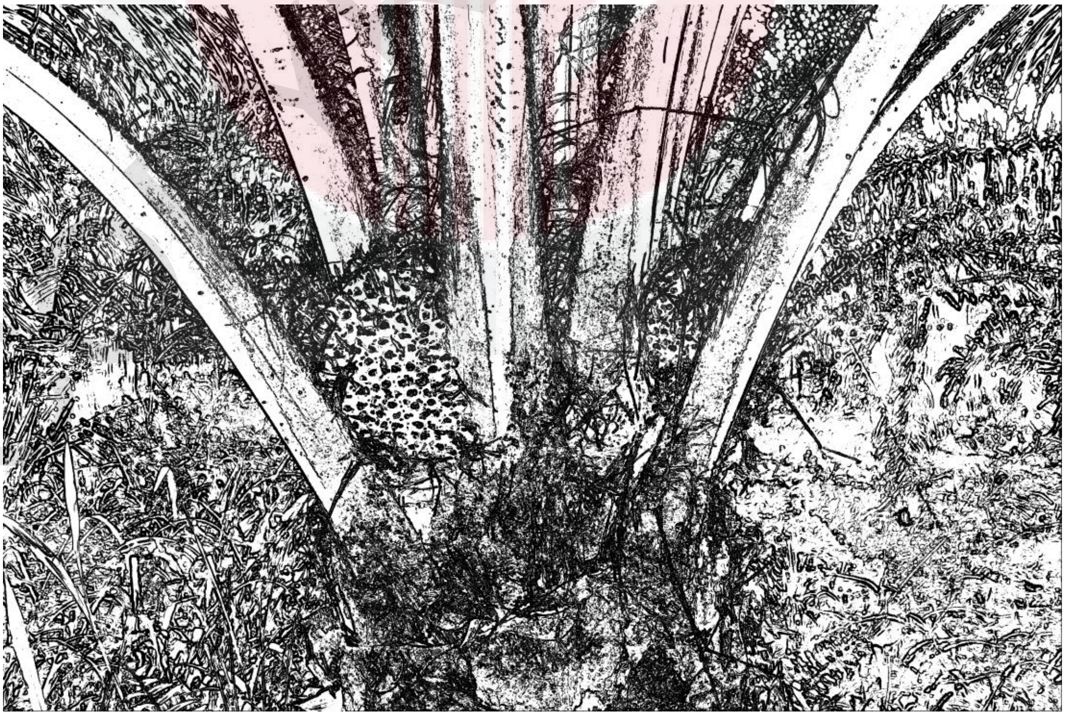


Figure 102: Final filtered image

TREE 7

Figure 103: Angle A Original image

Method 1 (Angle A)

Figure 104: Final binarized masked image

Method 2 (Angle A)

Figure 105: Final detected image

Method 3 (Angle A)

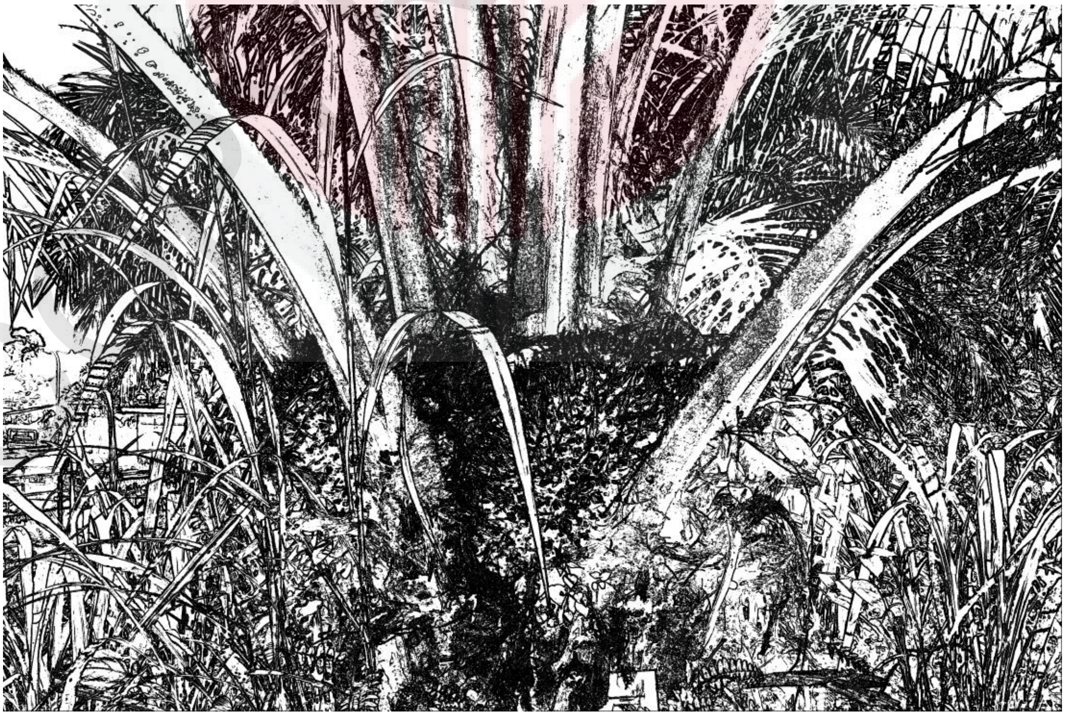


Figure 106: Final filtered image

Figure 107: Angle B Original image

Method 1 (Angle B)

Figure 108: Final binarized masked image

Method 2 (Angle B)

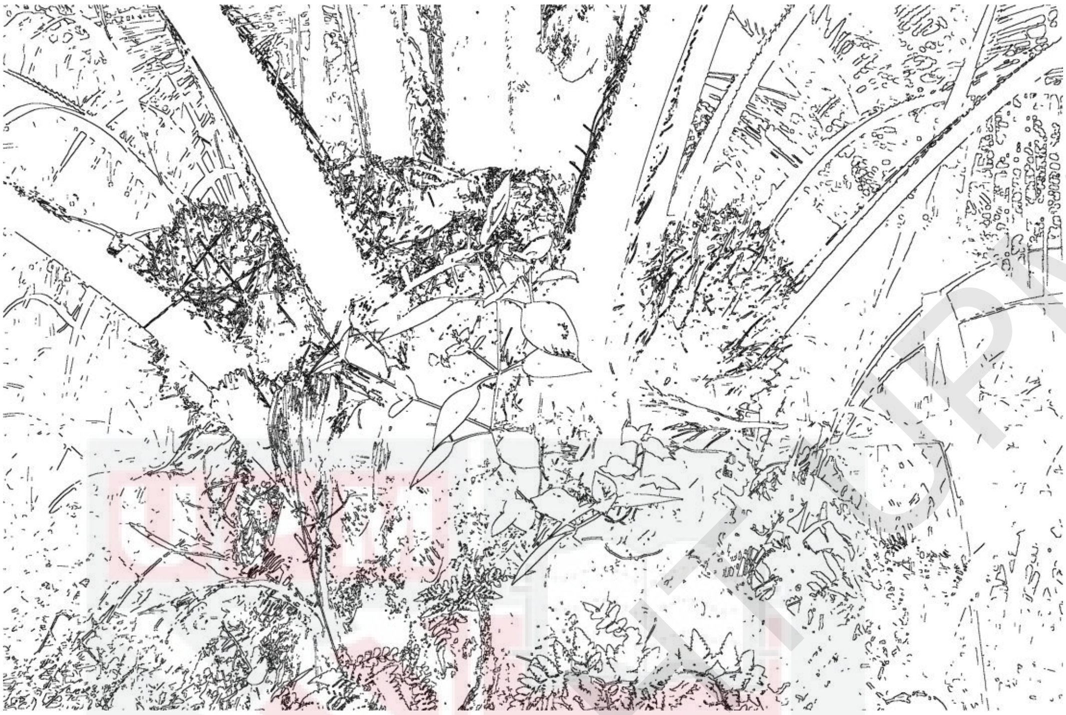


Figure 109: Final detected image

Method 3 (Angle B)

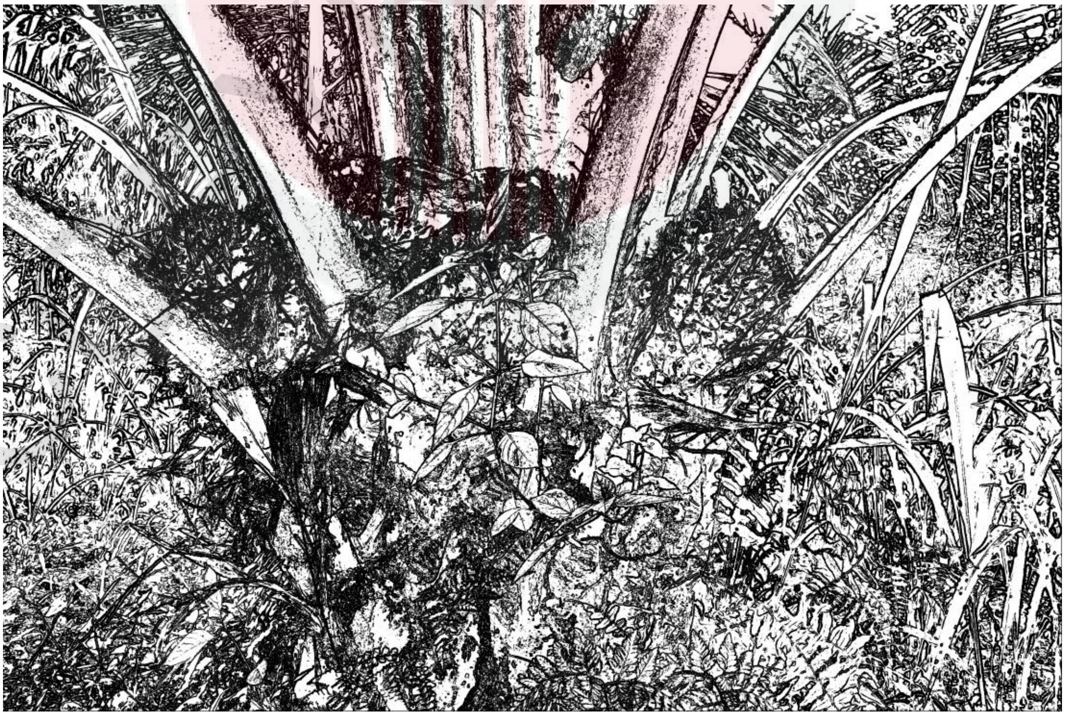


Figure 110: Final filtered image

Figure 111: Angle C Original image

Method 1 (Angle C)

Figure 112: Final binarized masked image

Method 2 (Angle C)

Figure 113: Final detected image

Method 3 (Angle C)

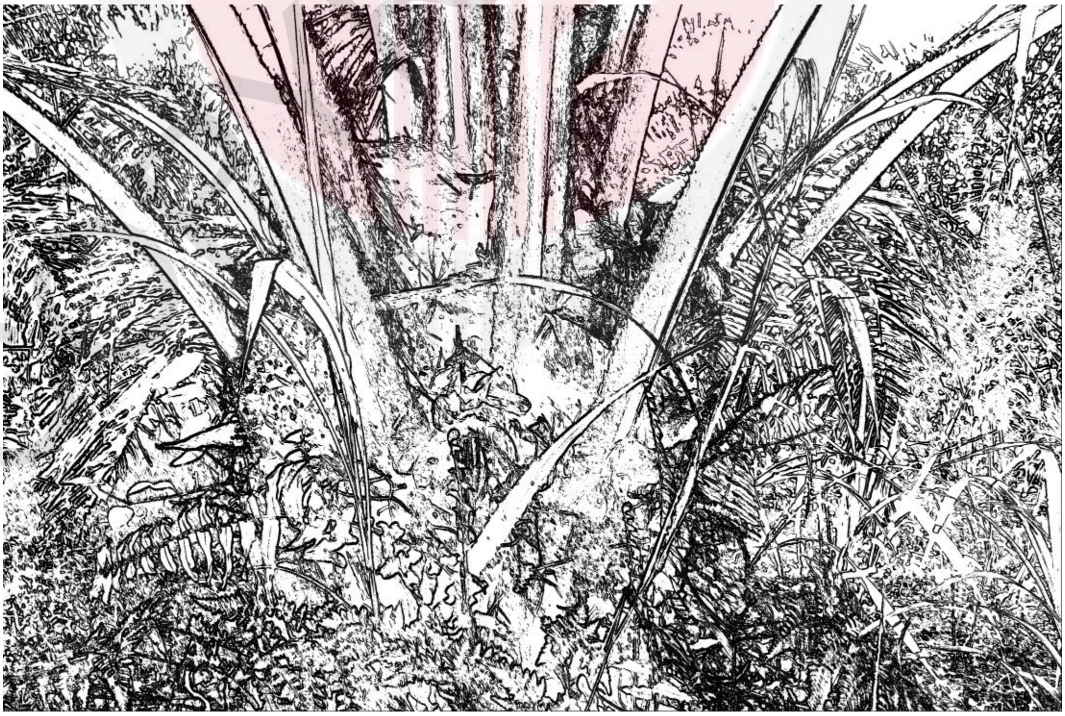


Figure 114: Final filtered image

Figure 115: Angle D Original image

Method 1 (Angle D)

Figure 116: Final binarized masked image

Method 2 (Angle D)

Figure 117: Final detected image

Method 3 (Angle D)

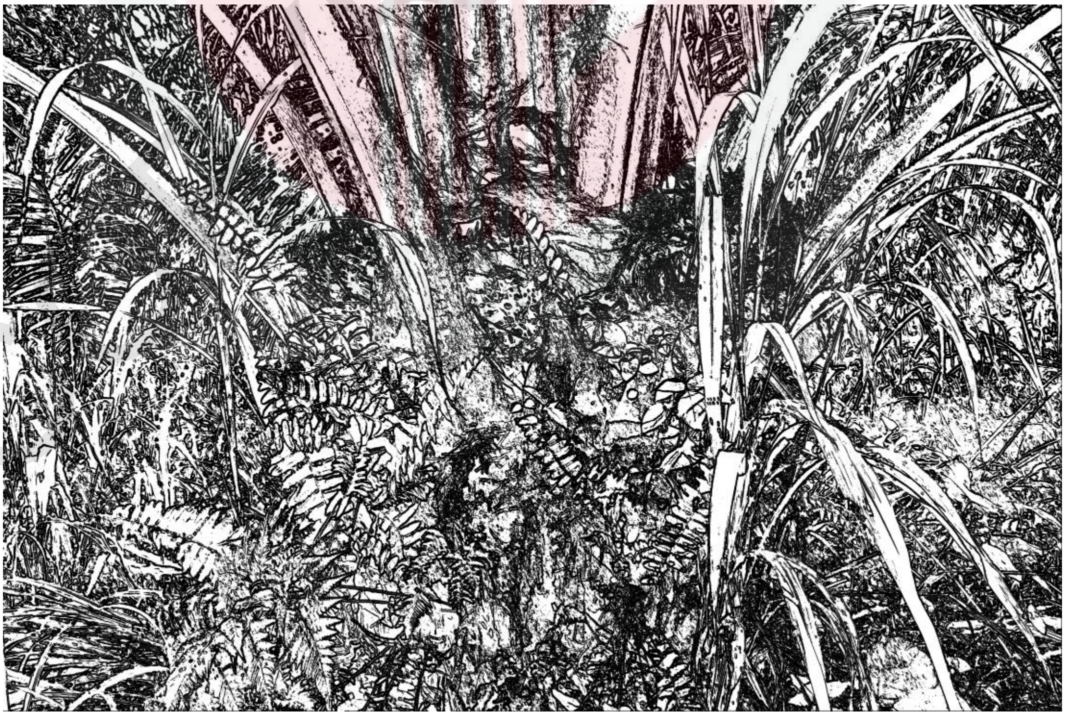


Figure 118: Final filtered image

TREE 8

Figure 119: Angle A Original image

Method 1 (Angle A)

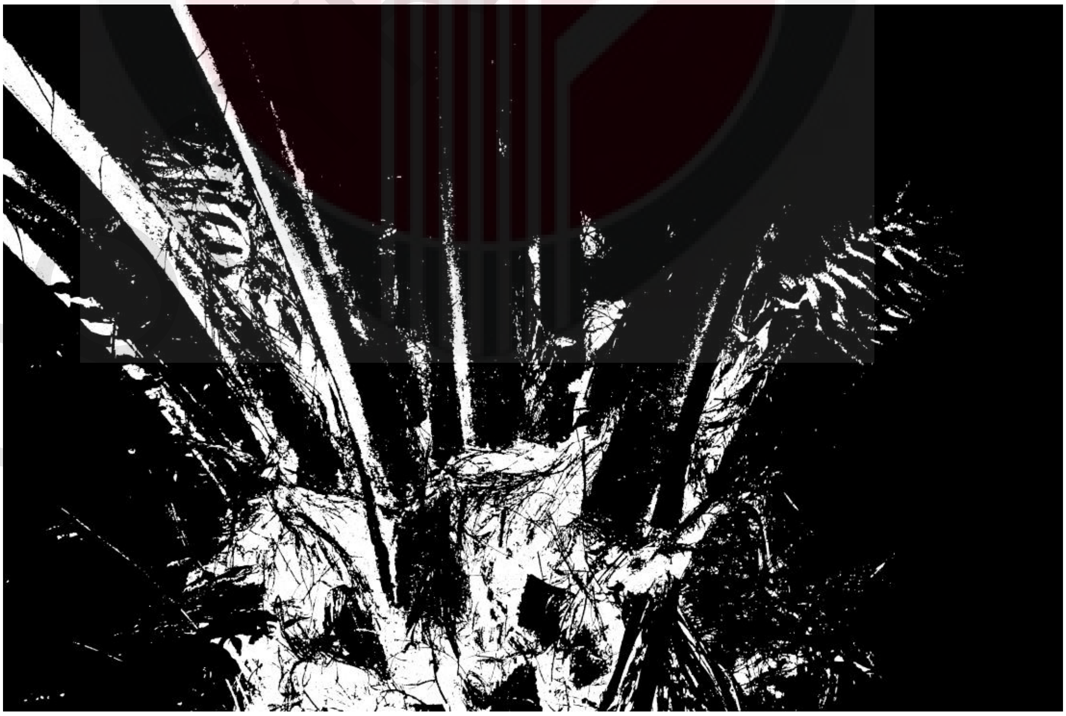


Figure 120: Final binarized masked image

Method 2 (Angle A)

Figure 121: Final detected image

Method 3 (Angle A)

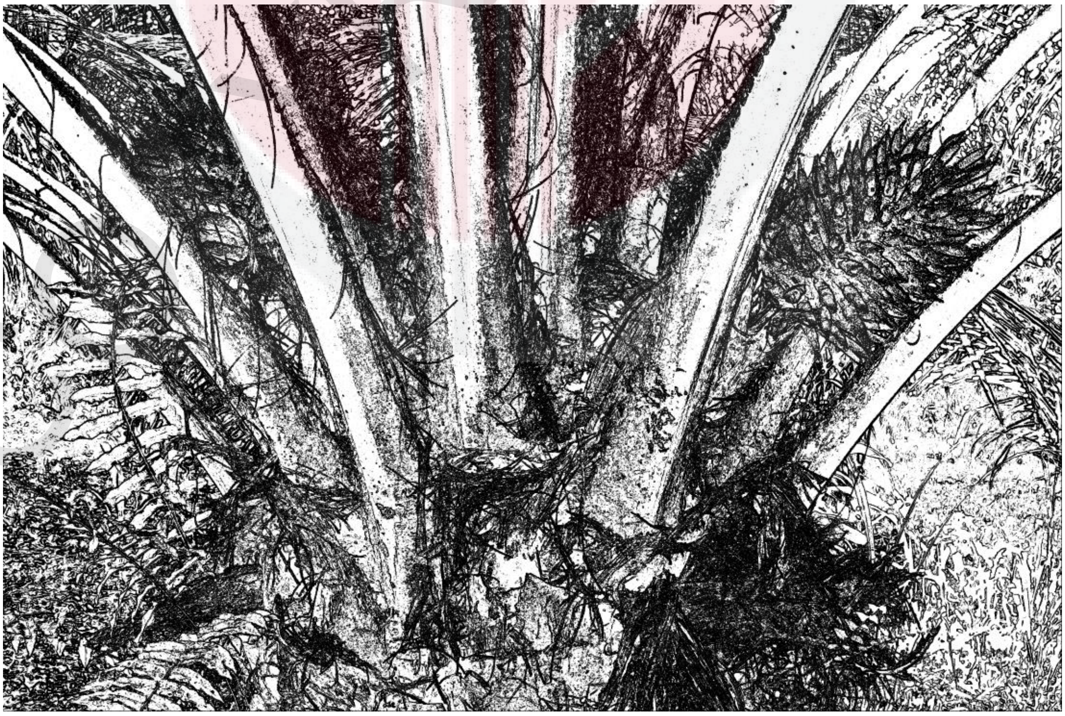


Figure 122: Final filtered image

Figure 123: Angle B Original image

Method 1 (Angle B)

Figure 124: Final binarized masked image

Method 2 (Angle B)

Figure 125: Final detected image

Method 3 (Angle B)

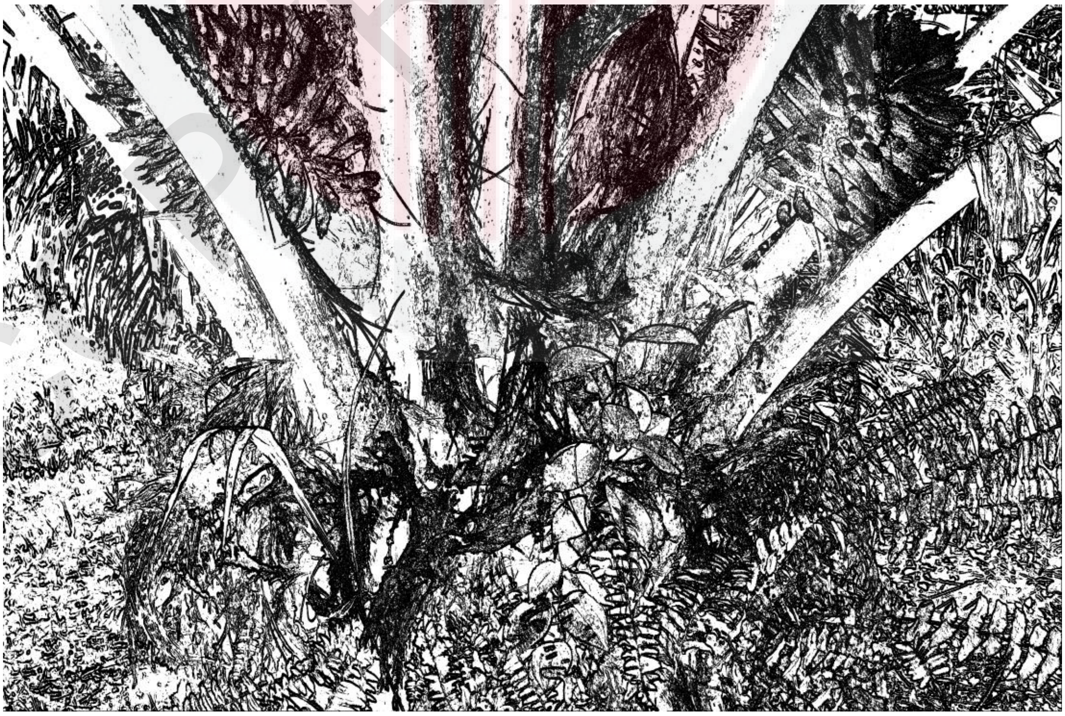


Figure 126: Final filtered image

Figure 127: Angle C Original image

Method 1 (Angle C)

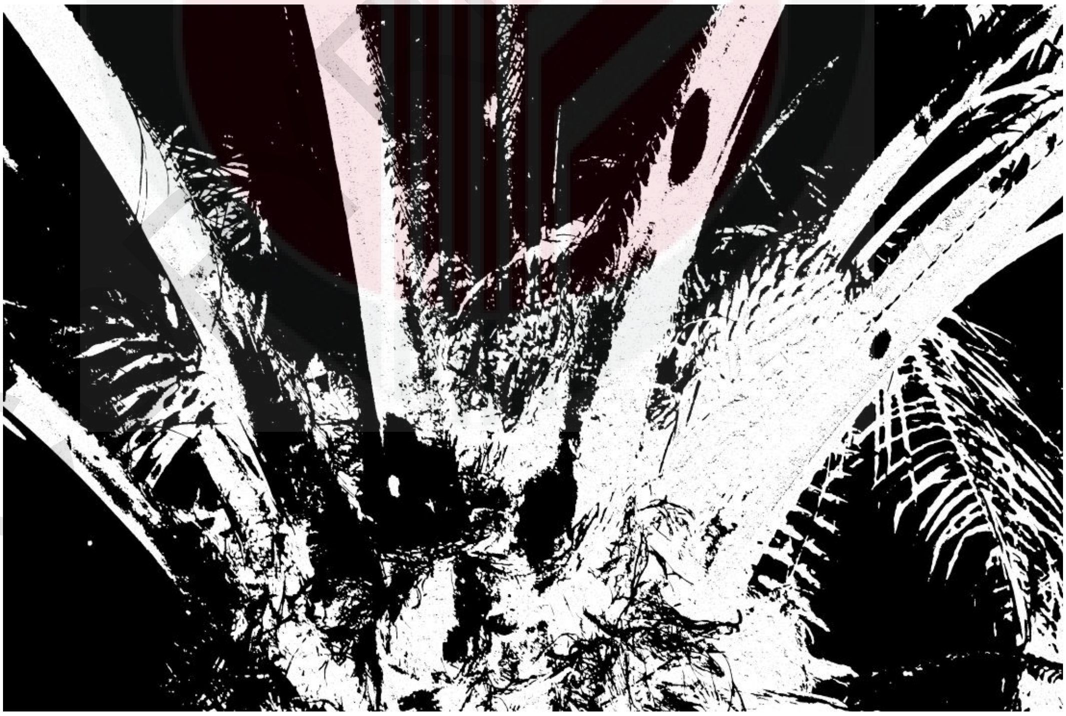


Figure 128: Final binarized masked image

Method 2 (Angle C)

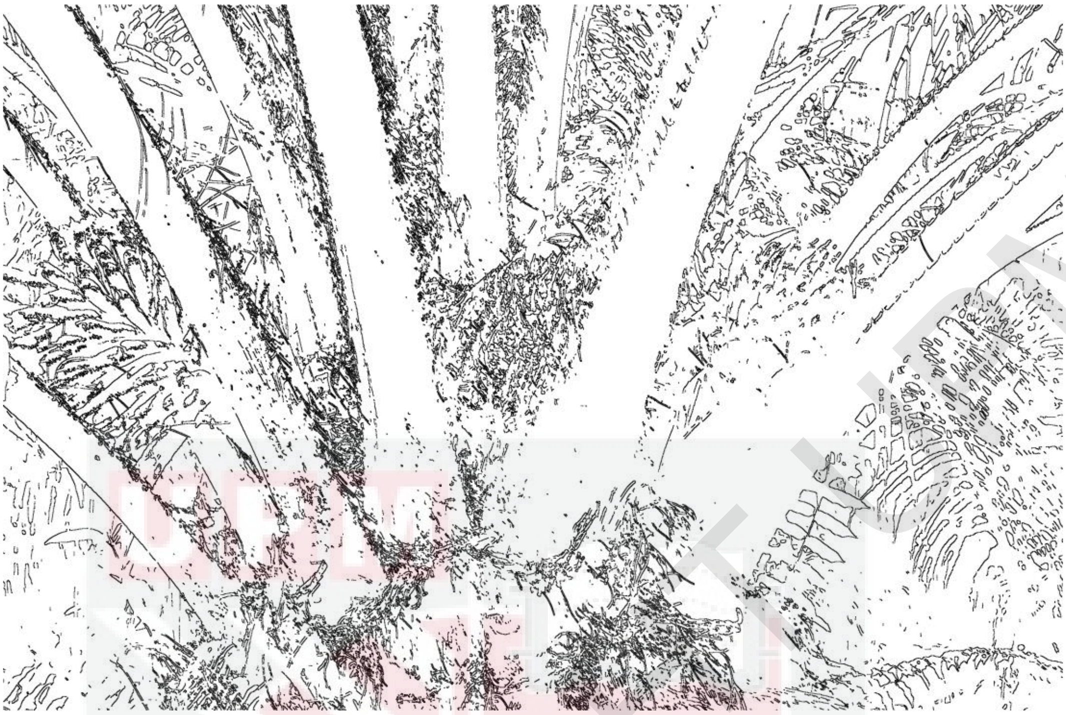


Figure 129: Final detected image

Method 3 (Angle C)

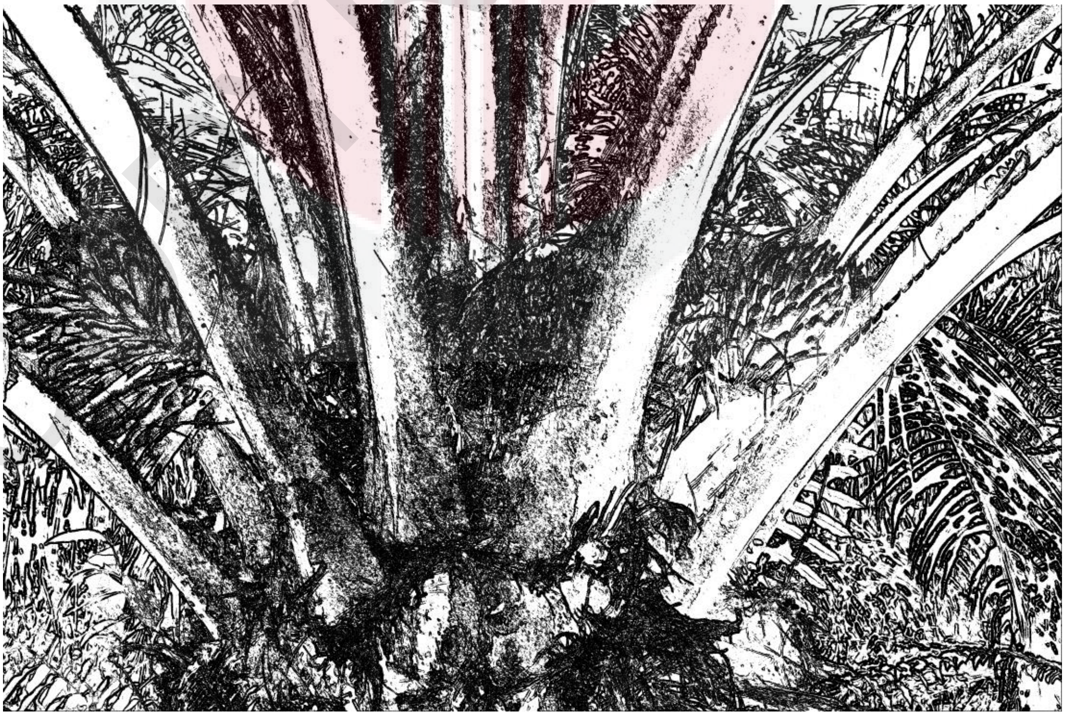


Figure 130: Final filtered image

Figure 131: Angle D Original image

Method 1 (Angle D)

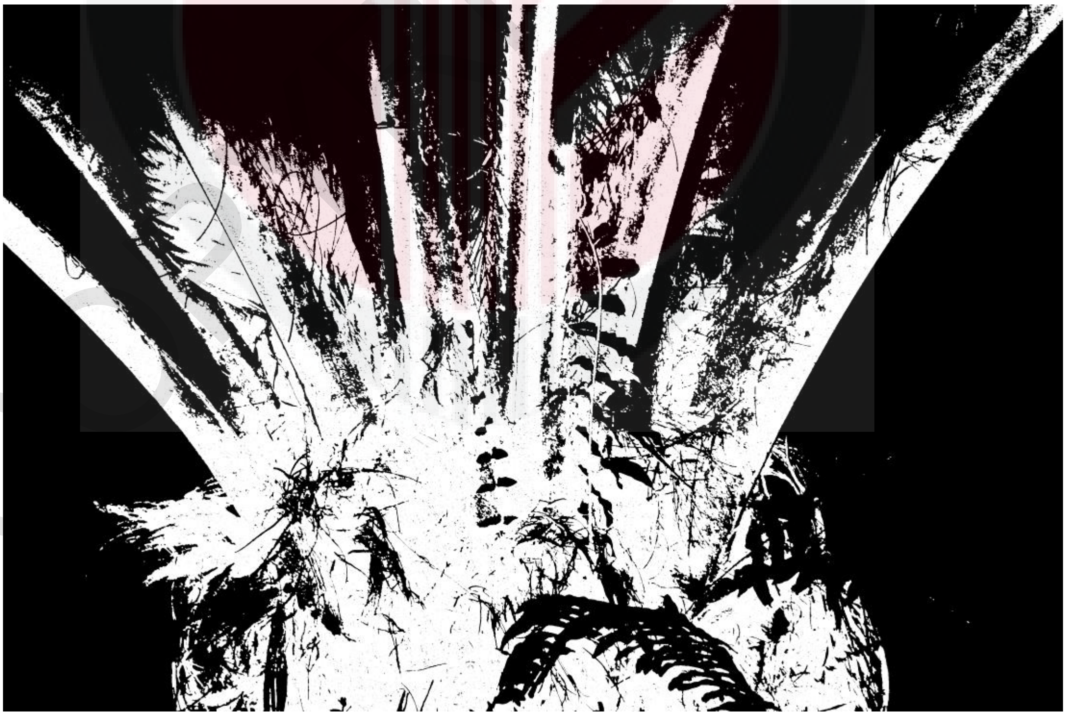


Figure 132: Final binarized masked image

Method 2 (Angle D)

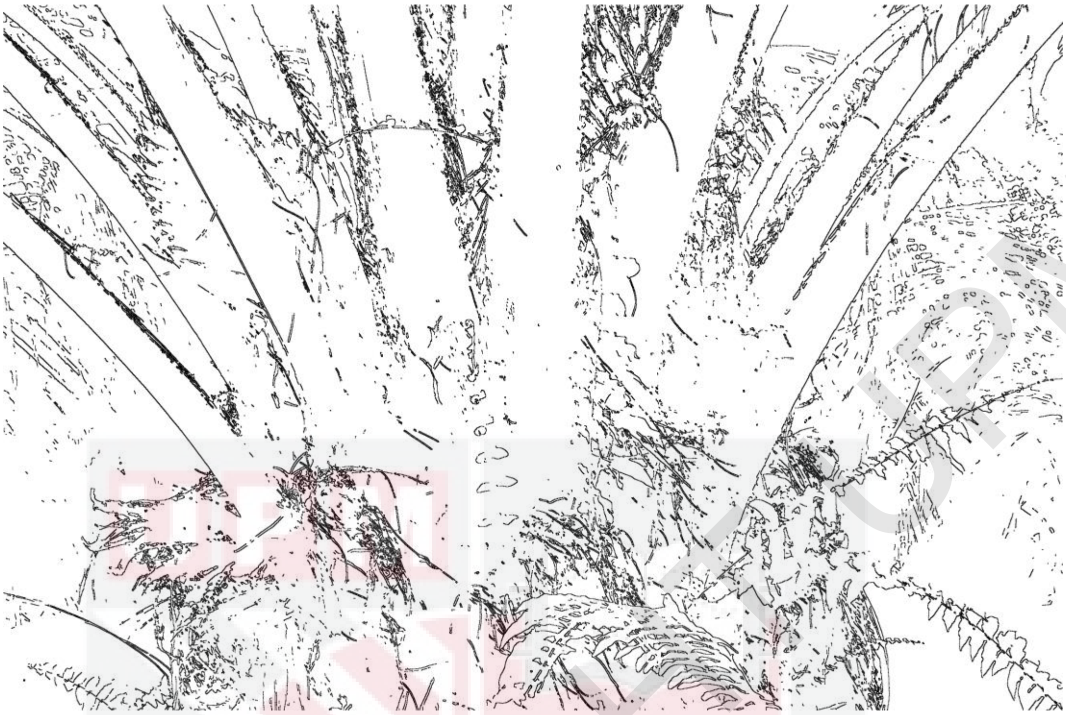


Figure 133: Final detected image

Method 3 (Angle D)

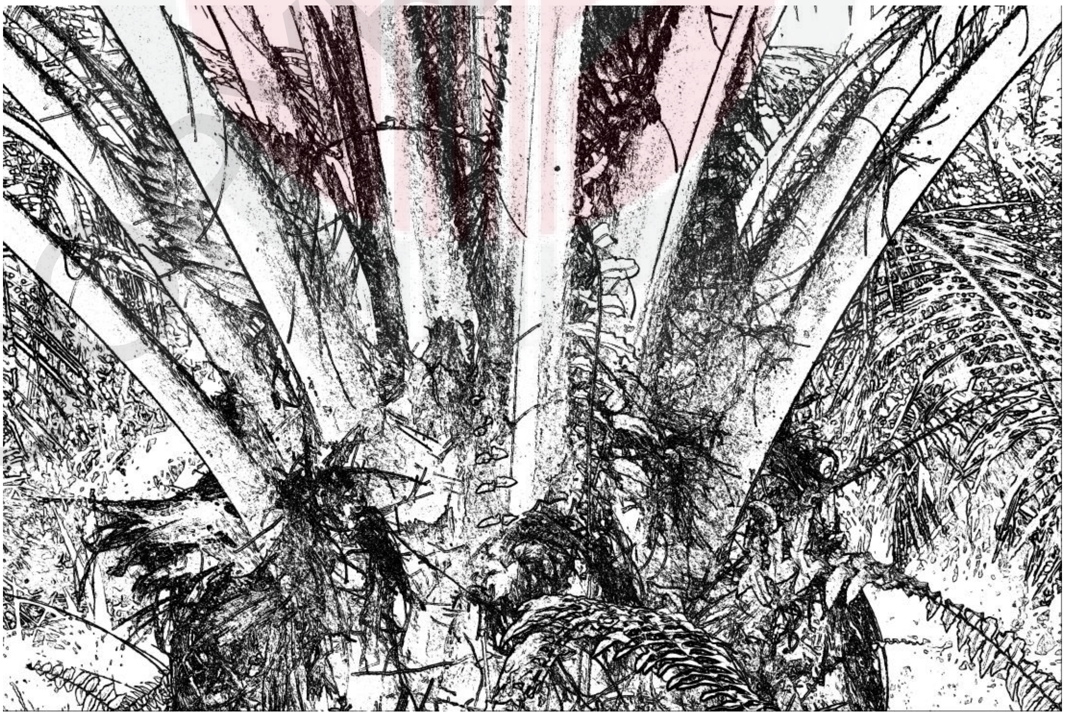


Figure 134: Final filtered image

TREE 9

Figure 135: Angle A Original image

Method 1 (Angle A)

Figure 136: Final binarized masked image

Method 2 (Angle A)

Figure 137: Final detected image

Method 3 (Angle A)

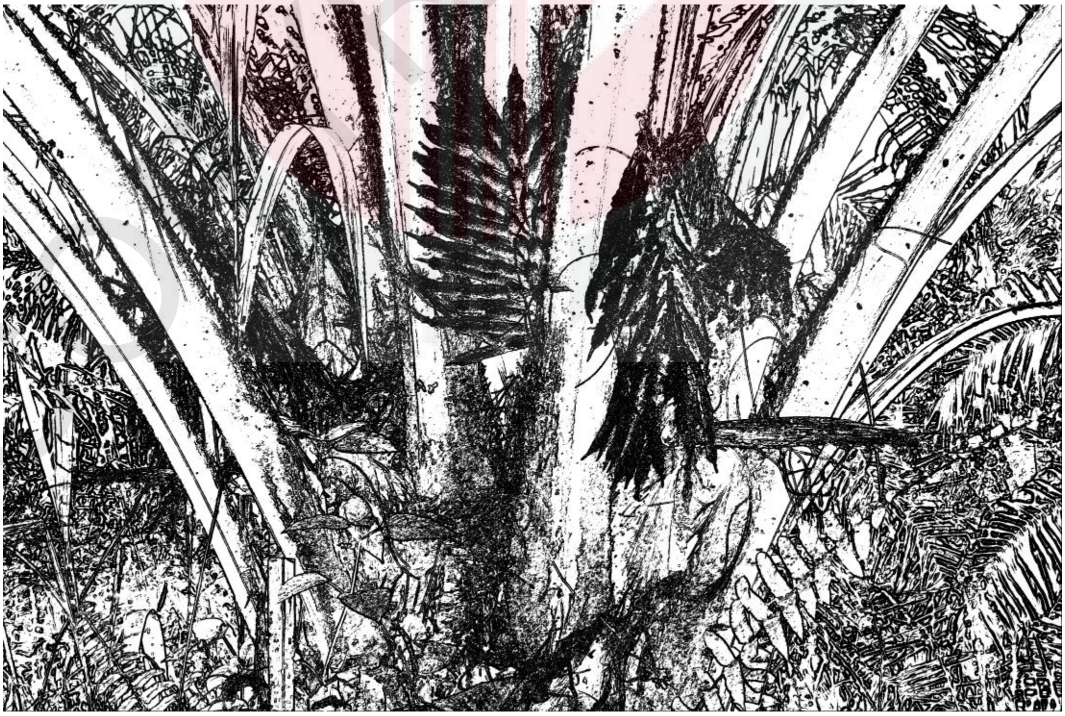


Figure 138: Final filtered image

Figure 139: Angle B Original image

Method 1 (Angle B)

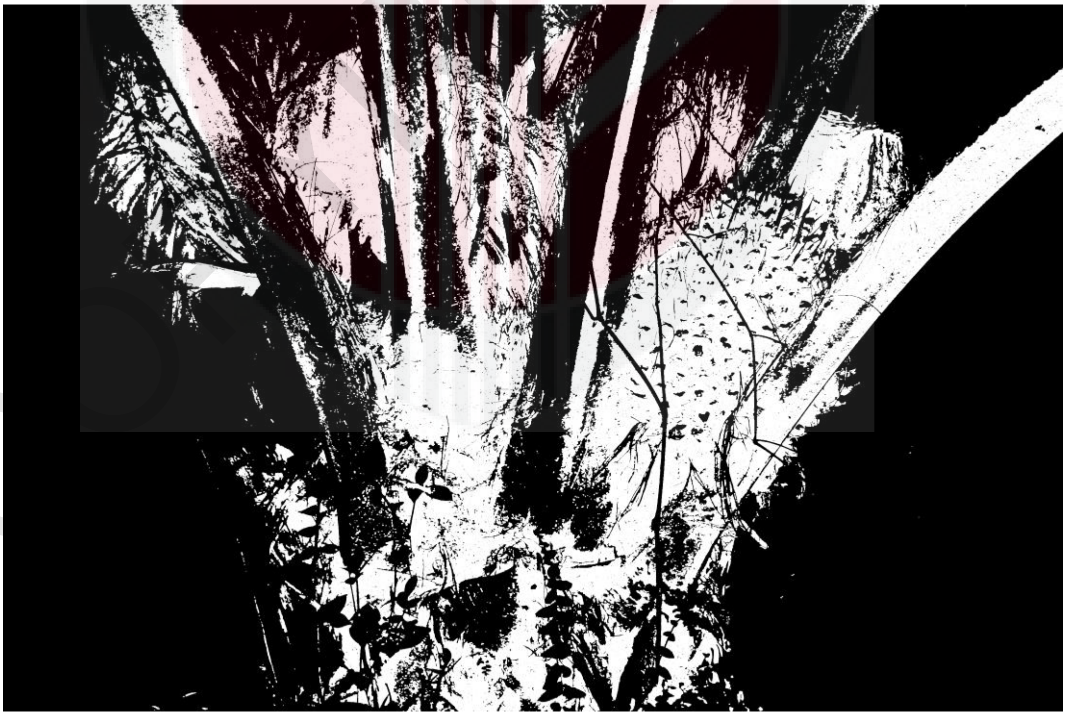


Figure 140: Final binarized masked image

Method 2 (Angle B)

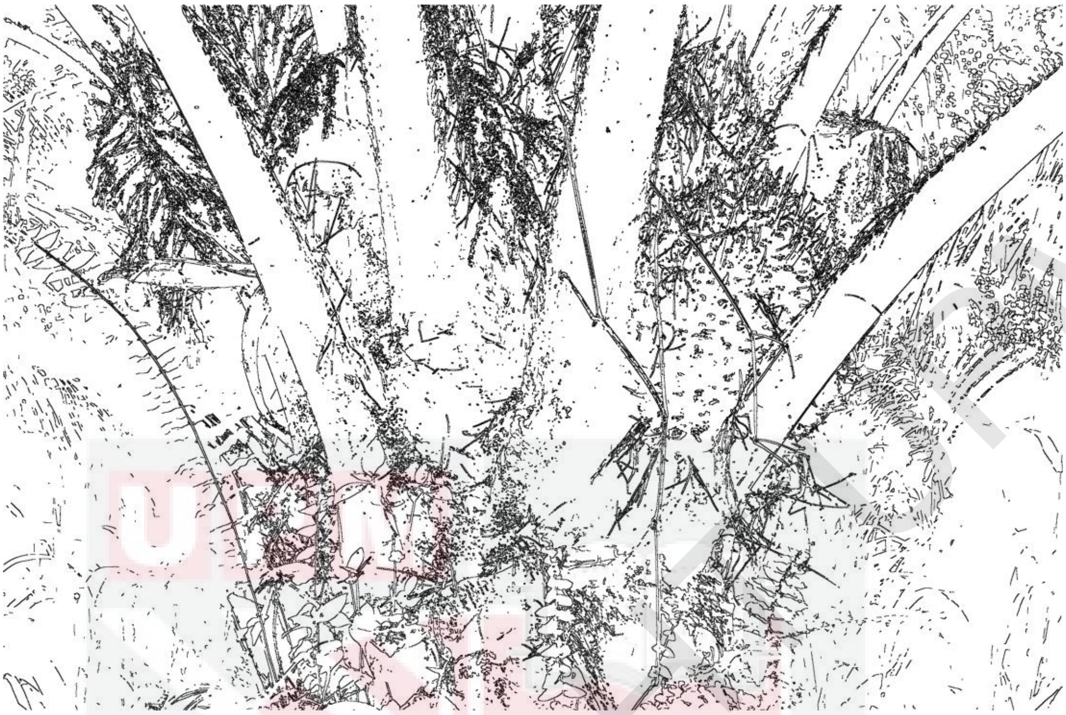


Figure 141: Final detected image

Method 3 (Angle B)

Figure 142: Final filtered image

Figure 143: Angle C Original image

Method 1 (Angle C)

Figure 144: Final binarized masked image

Method 2 (Angle C)

Figure 145: Final detected image

Method 3 (Angle C)

Figure 146: Final filtered image

Figure 147: Angle D Original image

Method 1 (Angle D)

Figure 148: Final binarized masked image

Method 2 (Angle D)

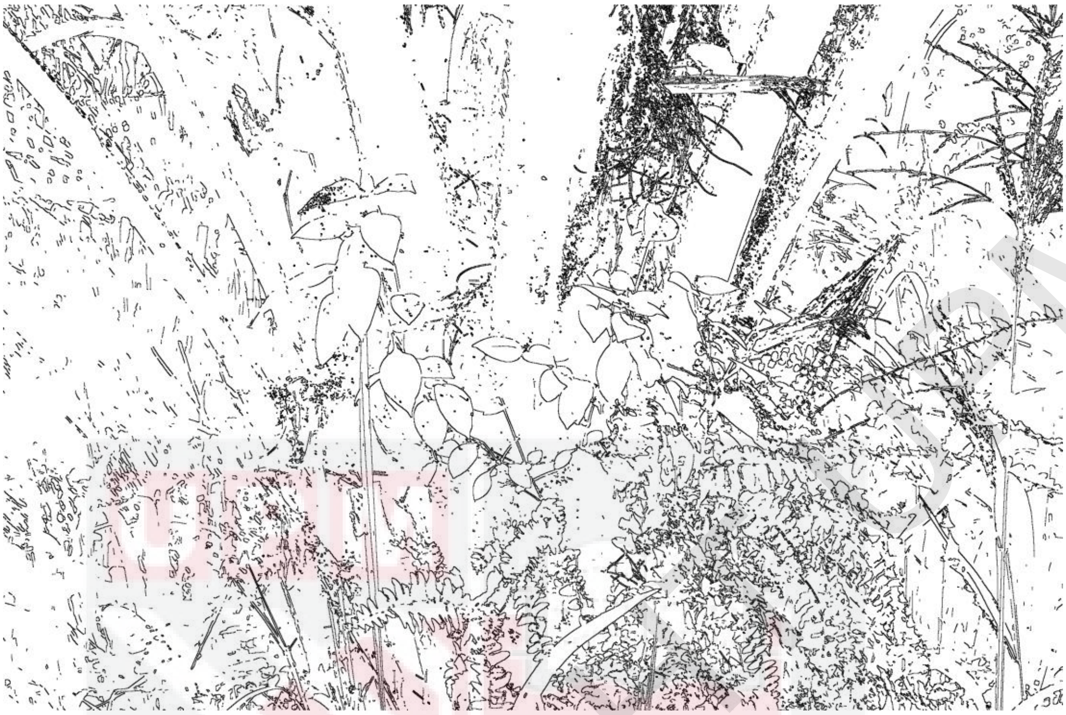


Figure 149: Final detected image

Method 3 (Angle D)

Figure 150: Final filtered image

TREE 10

Figure 151: Angle A Original image

Method 1 (Angle A)

Figure 152: Final binarized masked image

Method 2 (Angle A)

Figure 153: Final detected image

Method 3 (Angle A)

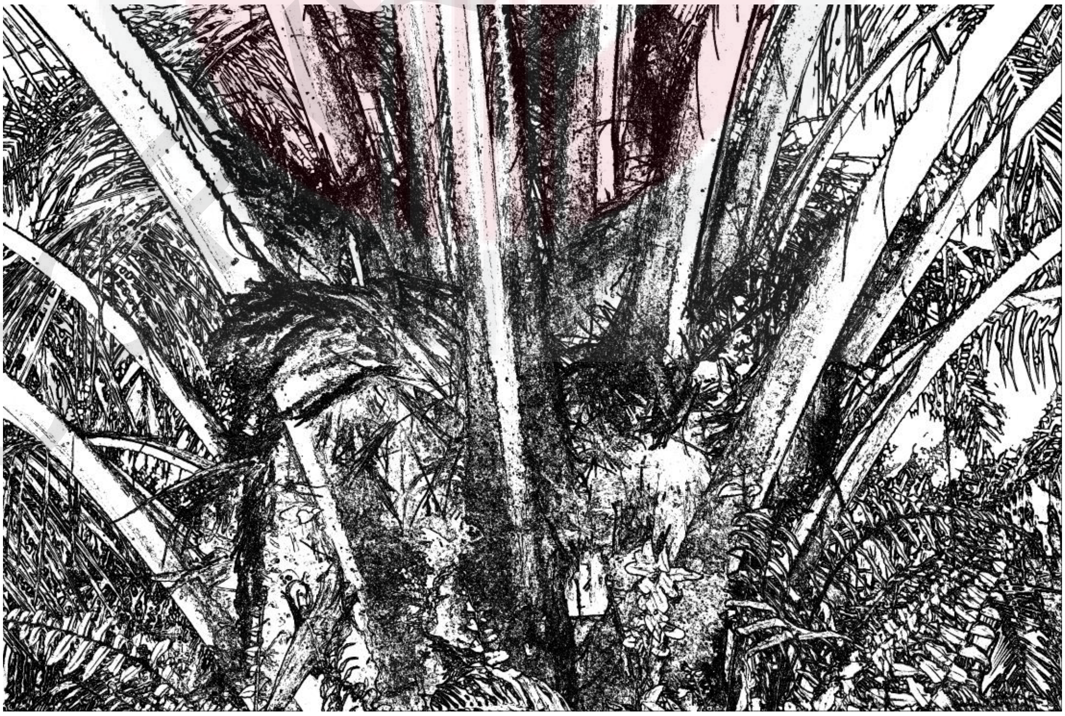


Figure 154: Final filtered image

Figure 155: Angle B Original image

Method 1 (Angle B)

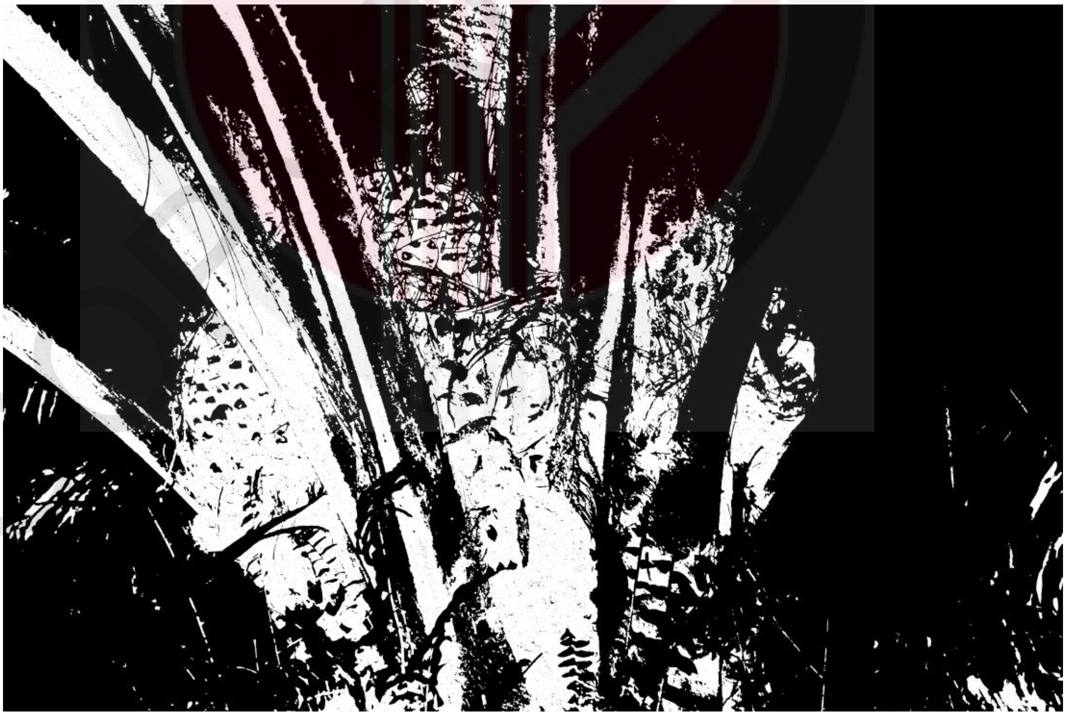


Figure 156: Final binarized masked image

Method 2 (Angle B)

Figure 157: Final detected image

Method 3 (Angle B)

Figure 158: Final filtered image

Figure 159: Angle C Original image

Method 1 (Angle C)

Figure 160: Final binarized masked image

Method 2 (Angle C)

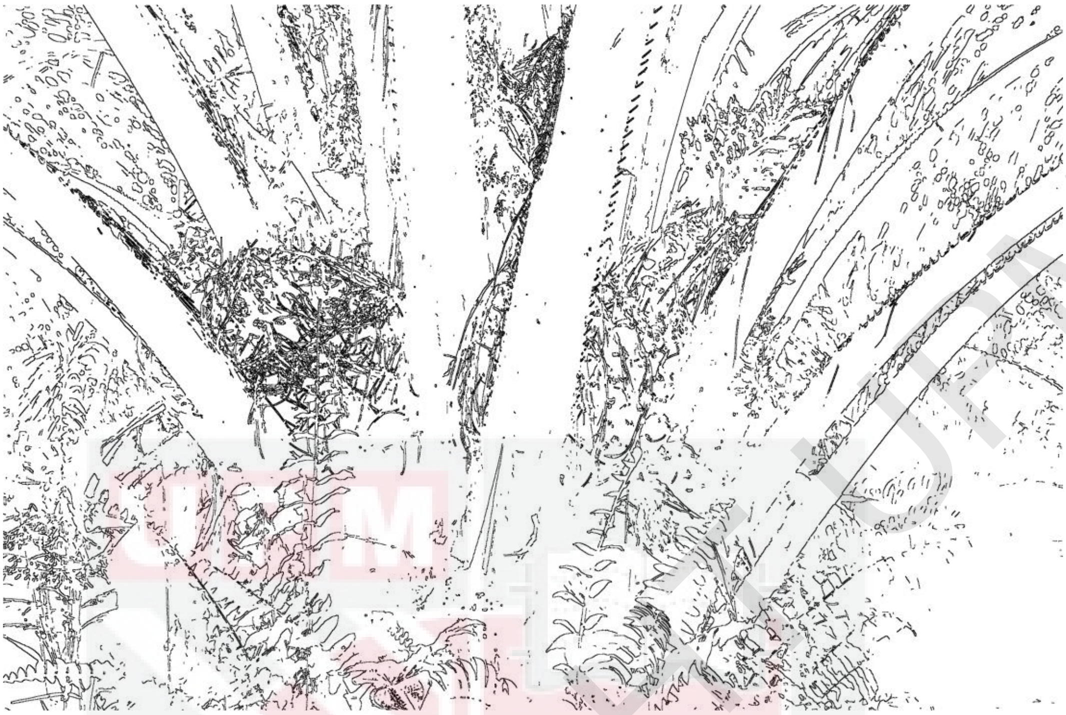


Figure 161: Final detected image

Method 3 (Angle C)

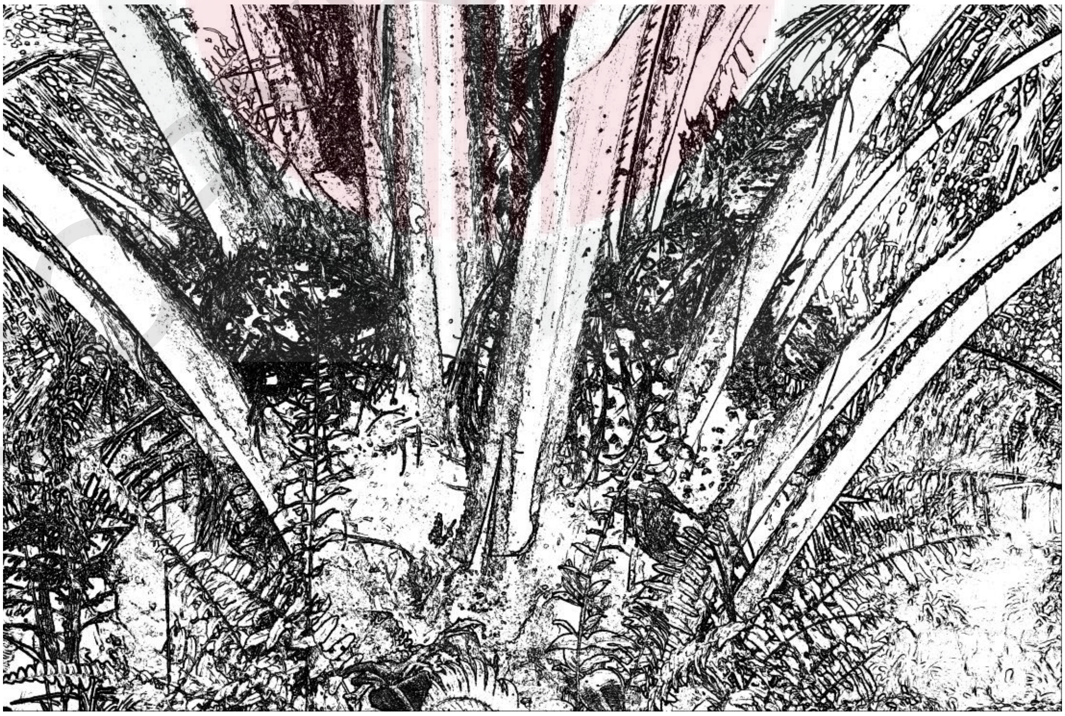


Figure 162: Final filtered image

Figure 163: Angle D Original image

Method 1 (Angle D)

Figure 164: Final binarized masked image

Method 2 (Angle D)

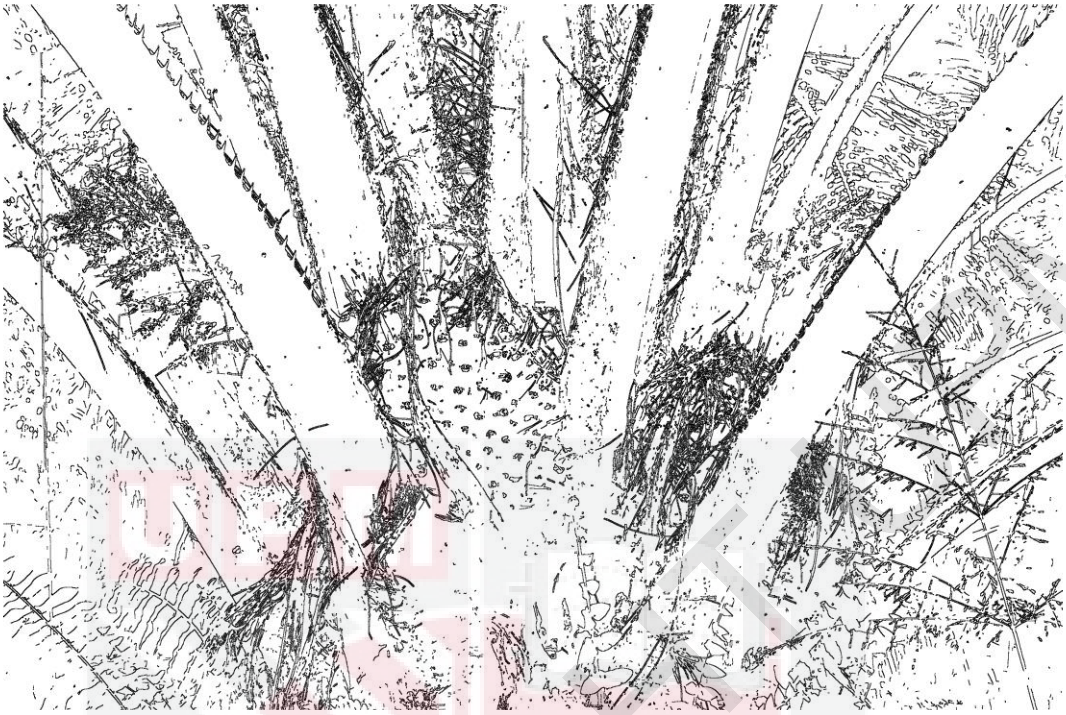


Figure 165: Final detected image

Method 3 (Angle D)

Figure 166: Final filtered image

MATLAB CODING FOR METHOD 1 (TREE 1: ANGLE A&B)

```
clear all, close all;
clc;

sawit = imread ('1A.jpg');
r = sawit(:,:,1);
g = sawit(:,:,2);
b = sawit(:,:,3);

mask = (r < 100) & (g < 100) & (b < 70);

maskedRgbImage = bsxfun(@times, sawit, cast(mask,class(sawit)));
figure, imshow(maskedRgbImage)
imwrite(maskedRgbImage,'1AcnyinvrtmaskedRgbImage.jpg')

se = strel('line',55,70)
background = imopen(maskedRgbImage,se);
imshow(background)
I2 = maskedRgbImage - background;
figure, imshow(I2)
imwrite(I2,'1AcnyinvrtI2.jpg')

C=double(I2);

for i=1:size(C,1)-2
    for j=1:size(C,2)-2
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        B(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end
figure,imshow(B); title('Sobel gradient');
imwrite(B,'1AcnyinvrtB.jpg')

sawit = imread ('1B.jpg');
r = sawit(:,:,1);
g = sawit(:,:,2);
b = sawit(:,:,3);
```

```
mask = (r < 100) & (g < 100) & (b < 70);

maskedRgbImage = bsxfun(@times, sawit, cast(mask,class(sawit)));
figure, imshow(maskedRgbImage)
imwrite(maskedRgbImage,'1BcnyinvrtmaskedRgbImage.jpg')

se = strel('line',55,70)
background = imopen(maskedRgbImage,se);
imshow(background)
I2 = maskedRgbImage - background;
figure, imshow(I2)
imwrite(I2,'1BcnyinvrtI2.jpg')

C=double(I2);

for i=1:size(C,1)-2
    for j=1:size(C,2)-2
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        B(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end
figure,imshow(B); title('Sobel gradient');
imwrite(B,'1BcnyinvrtB.jpg')
```

MATLAB CODING FOR METHOD 2 (TREE 1: ANGLE A&B)

```
clear all, close all;
clc;

%Input image
img = imread ('1A.jpg');
%Show input image
%%%figure, imshow(img);
img = rgb2gray(img);
img = double (img);
%Value for Thresholding
T_Low = 0.075;
T_High = 0.175;
%Gaussian Filter Coefficient
B = [2, 4, 5, 4, 2; 4, 9, 12, 9, 4;5, 12, 15, 12, 5;4, 9, 12, 9,
4;2, 4, 5, 4, 2 ];
B = 1/159.* B;
%Convolution of image by Gaussian Coefficient
A=conv2(img, B, 'same');
%Filter for horizontal and vertical direction
KGx = [-1, 0, 1; -2, 0, 2; -1, 0, 1];
KGy = [1, 2, 1; 0, 0, 0; -1, -2, -1];
%Convolution by image by horizontal and vertical filter
Filtered_X = conv2(A, KGx, 'same');
Filtered_Y = conv2(A, KGy, 'same');
%Calculate directions/orientations
direction = atan2 (Filtered_Y, Filtered_X);
direction = direction*180/pi;
length=size(A,1);
width=size(A,2);
%Adjustment for negative directions, making all directions positive
for i=1:length
    for j=1:width
        if (direction(i,j)<0)
            direction(i,j)=360+direction(i,j);
        end;
    end;
end;
direction2=zeros(length, width);
%Adjusting directions to nearest 0, 45, 90, or 135 degree
for i = 1  : length
    for j = 1 : width
        if ((direction(i, j) >= 0 ) && (direction(i, j) < 22.5) ||
(direction(i, j) >= 157.5) && (direction(i, j) < 202.5) ||
(direction(i, j) >= 337.5) && (direction(i, j) <= 360))
            direction2(i, j) = 0;
        elseif ((direction(i, j) >= 22.5) && (direction(i, j) <
67.5) || (direction(i, j) >= 202.5) && (direction(i, j) < 247.5))
```

```
            direction2(i, j) = 45;
        elseif ((direction(i, j) >= 67.5 && direction(i, j) < 112.5)
|| (direction(i, j) >= 247.5 && direction(i, j) < 292.5))
            direction2(i, j) = 90;
        elseif ((direction(i, j) >= 112.5 && direction(i, j) <
157.5) || (direction(i, j) >= 292.5 && direction(i, j) < 337.5))
            direction2(i, j) = 135;
        end;
    end;
end;
%%%figure, imagesc(direction2); colorbar;
%Calculate magnitude
magnitude = (Filtered_X.^2) + (Filtered_Y.^2);
magnitude2 = sqrt(magnitude);
BW = zeros (length, width);
%Non-Maximum Supression
for i=2:length-1
    for j=2:width-1
        if (direction2(i,j)==0)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i,j+1), magnitude2(i,j-1)]));
        elseif (direction2(i,j)==45)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i+1,j-1), magnitude2(i-1,j+1)]));
        elseif (direction2(i,j)==90)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i+1,j), magnitude2(i-1,j)]));
        elseif (direction2(i,j)==135)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i+1,j+1), magnitude2(i-1,j-1)]));
        end;
    end;
end;
BW = BW.*magnitude2;
figure, imshow(BW);
imwrite(BW,'1Armnz2BWi.jpg')

%Hysteresis Thresholding
T_Low = T_Low * max(max(BW));
T_High = T_High * max(max(BW));
T_res = zeros (length, width);
for i = 1  : length
    for j = 1 : width
        if (BW(i, j) < T_Low)
            T_res(i, j) = 0;
        elseif (BW(i, j) > T_High)
            T_res(i, j) = 1;
        %Using 8-connected components
```

```
        elseif ( BW(i+1,j)>T_High || BW(i-1,j)>T_High ||
BW(i,j+1)>T_High || BW(i,j-1)>T_High || BW(i-1, j-1)>T_High || BW(i-
1, j+1)>T_High || BW(i+1, j+1)>T_High || BW(i+1, j-1)>T_High)
            T_res(i,j) = 1;
        end;
    end;
end;
edge_final = uint8(T_res.*255);
%Show final edge detection result
figure, imshow(edge_final);
imwrite(edge_final,'1Armnz2edge_finalii.jpg')

se = strel('line',55,85)
background = imopen(edge_final,se);
%%%imshow(background)
I2 = edge_final - background;
figure, imshow(I2)
imwrite(I2,'1Armnz2I2iii.jpg')

C=double(I2);

for i=1:size(C,1)-2
    for j=1:size(C,2)-2
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        B(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end
figure,imshow(B); title('Sobel gradient');
imwrite(B,'1Armnz2Biv.jpg')

imgbw = im2bw(B);
%take complement to adhere to convention
imgbw = imcomplement(imgbw) ;

figure, imshow(imgbw)
imwrite(imgbw,'1Aimgbwrmnz2finalv.jpg')

%Input image
img = imread ('1B.jpg');
```

```
%Show input image
%%%figure, imshow(img);
img = rgb2gray(img);
img = double (img);
%Value for Thresholding
T_Low = 0.075;
T_High = 0.175;
%Gaussian Filter Coefficient
B = [2, 4, 5, 4, 2; 4, 9, 12, 9, 4;5, 12, 15, 12, 5;4, 9, 12, 9,
4;2, 4, 5, 4, 2 ];
B = 1/159.* B;
%Convolution of image by Gaussian Coefficient
A=conv2(img, B, 'same');
%Filter for horizontal and vertical direction
KGx = [-1, 0, 1; -2, 0, 2; -1, 0, 1];
KGy = [1, 2, 1; 0, 0, 0; -1, -2, -1];
%Convolution by image by horizontal and vertical filter
Filtered_X = conv2(A, KGx, 'same');
Filtered_Y = conv2(A, KGy, 'same');
%Calculate directions/orientations
direction = atan2 (Filtered_Y, Filtered_X);
direction = direction*180/pi;
length=size(A,1);
width=size(A,2);
%Adjustment for negative directions, making all directions positive
for i=1:length
    for j=1:width
        if (direction(i,j)<0)
            direction(i,j)=360+direction(i,j);
        end;
    end;
end;
direction2=zeros(length, width);
%Adjusting directions to nearest 0, 45, 90, or 135 degree
for i = 1   : length
    for j = 1 : width
        if ((direction(i, j) >= 0 ) && (direction(i, j) < 22.5) ||
(direction(i, j) >= 157.5) && (direction(i, j) < 202.5) ||
(direction(i, j) >= 337.5) && (direction(i, j) <= 360))
            direction2(i, j) = 0;
        elseif ((direction(i, j) >= 22.5) && (direction(i, j) <
67.5) || (direction(i, j) >= 202.5) && (direction(i, j) < 247.5))
            direction2(i, j) = 45;
        elseif ((direction(i, j) >= 67.5 && direction(i, j) < 112.5)
|| (direction(i, j) >= 247.5 && direction(i, j) < 292.5))
            direction2(i, j) = 90;
        elseif ((direction(i, j) >= 112.5 && direction(i, j) <
157.5) || (direction(i, j) >= 292.5 && direction(i, j) < 337.5))
            direction2(i, j) = 135;
        end;
```

```
    end;
end;
%%%figure, imagesc(direction2); colorbar;
%Calculate magnitude
magnitude = (Filtered_X.^2) + (Filtered_Y.^2);
magnitude2 = sqrt(magnitude);
BW = zeros (length, width);
%Non-Maximum Supression
for i=2:length-1
    for j=2:width-1
        if (direction2(i,j)==0)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i,j+1), magnitude2(i,j-1)]));
        elseif (direction2(i,j)==45)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i+1,j-1), magnitude2(i-1,j+1)]));
        elseif (direction2(i,j)==90)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i+1,j), magnitude2(i-1,j)]));
        elseif (direction2(i,j)==135)
            BW(i,j) = (magnitude2(i,j) == max([magnitude2(i,j),
magnitude2(i+1,j+1), magnitude2(i-1,j-1)]));
        end;
    end;
end;
BW = BW.*magnitude2;
figure, imshow(BW);
imwrite(BW,'1Brmnz2BWi.jpg')

%Hysteresis Thresholding
T_Low = T_Low * max(max(BW));
T_High = T_High * max(max(BW));
T_res = zeros (length, width);
for i = 1  : length
    for j = 1 : width
        if (BW(i, j) < T_Low)
            T_res(i, j) = 0;
        elseif (BW(i, j) > T_High)
            T_res(i, j) = 1;
        %Using 8-connected components
        elseif ( BW(i+1,j)>T_High || BW(i-1,j)>T_High ||
BW(i,j+1)>T_High || BW(i,j-1)>T_High || BW(i-1, j-1)>T_High || BW(i-
1, j+1)>T_High || BW(i+1, j+1)>T_High || BW(i+1, j-1)>T_High)
            T_res(i,j) = 1;
        end;
    end;
end;
edge_final = uint8(T_res.*255);
%Show final edge detection result
figure, imshow(edge_final);
```

```
imwrite(edge_final,'1Brmnz2edge_finalii.jpg')

se = strel('line',55,85)
background = imopen(edge_final,se);
%%%imshow(background)
I2 = edge_final - background;
figure, imshow(I2)
imwrite(I2,'1Brmnz2I2iii.jpg')

C=double(I2);

for i=1:size(C,1)-2
    for j=1:size(C,2)-2
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        B(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end
figure,imshow(B); title('Sobel gradient');
imwrite(B,'1Brmnz2Biv.jpg')

imgbw = im2bw(B);
%take complement to adhere to convention
imgbw = imcomplement(imgbw) ;

figure, imshow(imgbw)
imwrite(imgbw,'1Bimgbwrmnz2finalv.jpg')
```

MATLAB CODING FOR METHOD 3 (TREE 1: ANGLE A&B)

```
clear all, close all, clc;

A = imread ('1A.jpg');
B = rgb2gray (A);

C=double(B);

for i=1:size(C,1)-2;
    for j=1:size(C,2)-2;
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        B(i,j)=sqrt(Gx.^2+Gy.^2);


    end
end
figure,imshow(B); title('Sobel gradient');
imwrite(B,'1AxB.jpg')

 BW=im2bw(B,0.1);
    figure, imshow(BW);
    title('BW Image') ;
    imwrite(BW,'1AxBW.jpg')

imgbw = imcomplement(BW) ;
figure, imshow(imgbw);
imwrite(imgbw,'1Aximgbw.jpg')

A = imread ('1B.jpg');
B = rgb2gray (A);

C=double(B);

for i=1:size(C,1)-2;
    for j=1:size(C,2)-2;
        %Sobel mask for x-direction:
        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-
(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        %Sobel mask for y-direction:
```

```
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-
(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        %The gradient of the image
        %B(i,j)=abs(Gx)+abs(Gy);
        B(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end
figure,imshow(B); title('Sobel gradient');
imwrite(B,'1BxB.jpg')

 BW=im2bw(B,0.1);
    figure, imshow(BW);
    title('BW Image') ;
    imwrite(BW,'1BxBW.jpg')

imgbw = imcomplement(BW) ;
figure, imshow(imgbw);
imwrite(imgbw,'1Bximgbw.jpg')
```