LAMPIRAN

Lampiran 1 Data Pengujian Nomor Benang Rayon $Ne_1$30

No	Panjang(yard)	Berat (g)	Ne1	Tex
1	120	2,147	30,1832	19,5473
2	120	2,156	30,0572	19,6292
3	120	2,145	30,2114	19,5291
4	120	2,155	30,0712	19,6201
5	120	2,151	30,1271	19,5837
6	120	2,164	29,9461	19,7021
7	120	2,164	29,9461	19,7021
8	120	2,175	29,7947	19,8022
9	120	2,157	30,0433	19,6383
10	120	2,158	30,0294	19,6474
11	120	2,162	29,9738	19,6839
12	120	2,173	29,8221	19,7840
13	120	2,152	30,1131	19,5928
14	120	2,159	30,0155	19,6565
15	120	2,16	30,0016	19,6656
16	120	2,152	30,1131	19,5928
17	120	2,151	30,1271	19,5837
18	120	2,161	29,9877	19,6747
19	120	2,156	30,0572	19,6292
20	120	2,145	30,2114	19,5291
21	120	2,166	29,9185	19,7203
22	120	2,147	30,1832	19,5473
23	120	2,169	29,8771	19,7476
24	120	2,155	30,0712	19,6201
25	120	2,177	29,7673	19,8204
Rata-rata			30,0259	19,6499
SD			0,1248	0,0818
CV			0,4155%	0,4162%

Lampiran 2 Data Pengujian Nomor Benang Rayon $Ne_1$30/2

No	Panjang(yard)	Berat (g)	Ne1	Tex
1	120	4,258	15,20	38,81
2	120	4,262	15,19	38,84
3	120	4,285	15,11	39,05
4	120	4,156	15,58	37,88
5	120	4,277	15,14	38,98
6	120	4,371	14,81	39,84
7	120	4,504	14,37	41,05
8	120	4,171	15,52	38,01
9	120	4,302	15,05	39,21
10	120	4,283	15,11	39,04
11	120	4,269	15,16	38,91
12	120	4,376	14,79	39,88
13	120	4,571	14,16	41,66
14	120	4,337	14,93	39,53
15	120	4,218	15,35	38,44
16	120	4,295	15,07	39,15
17	120	4,28	15,12	39,01
18	120	4,324	14,97	39,41
19	120	4,248	15,24	38,72
20	120	4,361	14,84	39,75
21	120	4,326	14,96	39,43
22	120	4,296	15,07	39,15
23	120	4,301	15,05	39,20
24	120	4,255	15,21	38,78
25	120	4,361	14,84	39,75
Rata-rata			15,03	39,25
SD			0,3030	0,8088
CV			2,0157%	2,0603%

Lampiran 3 Data Pengukuran Metode Konvensional

No	$Ne_1$30	$Ne_1$30/2
	Diameter (mm)	
1	0,1579	0,2225
2	0,1582	0,2226
3	0,1578	0,2232
4	0,1582	0,2198
5	0,1580	0,2230
6	0,1585	0,2254
7	0,1585	0,2288
8	0,1589	0,2202
9	0,1583	0,2236
10	0,1583	0,2231
11	0,1585	0,2228
12	0,1589	0,2255
13	0,1581	0,2305
14	0,1583	0,2245
15	0,1584	0,2214
16	0,1581	0,2235
17	0,1580	0,2231
18	0,1584	0,2242
19	0,1582	0,2222
20	0,1578	0,2252
21	0,1586	0,2243
22	0,1579	0,2235
23	0,1587	0,2236
24	0,1582	0,2224
25	0,1590	0,2252
Rata-rata	0,158315	0,2237
SD	0,0003	0,0022
CV	0,2080%	1,0243%

Lampiran 4 Data Pengukuran Metode *Image Processing*

No	$Ne_1$30	$Ne_1$30/2
	Diameter (mm)	
1	0,1463	0,2134
2	0,1533	0,2161
3	0,1646	0,2251
4	0,1592	0,2213
5	0,1655	0,2234
6	0,1454	0,2238
7	0,1412	0,2248
8	0,1647	0,2168
9	0,1528	0,2265
10	0,1642	0,2267
11	0,1511	0,2272
12	0,1627	0,2275
13	0,1490	0,2332
14	0,1496	0,234
15	0,1468	0,2346
16	0,1531	0,2359
17	0,1650	0,2378
18	0,1622	0,2384
19	0,1483	0,218
20	0,1598	0,2421
21	0,1533	0,2423
22	0,1495	0,243
23	0,1570	0,2451
24	0,1667	0,2478
25	0,1538	0,2263
Rata-rata	0,1554	0,23
SD	0,0075	0,0097
CV	94,8554%	4,2292%

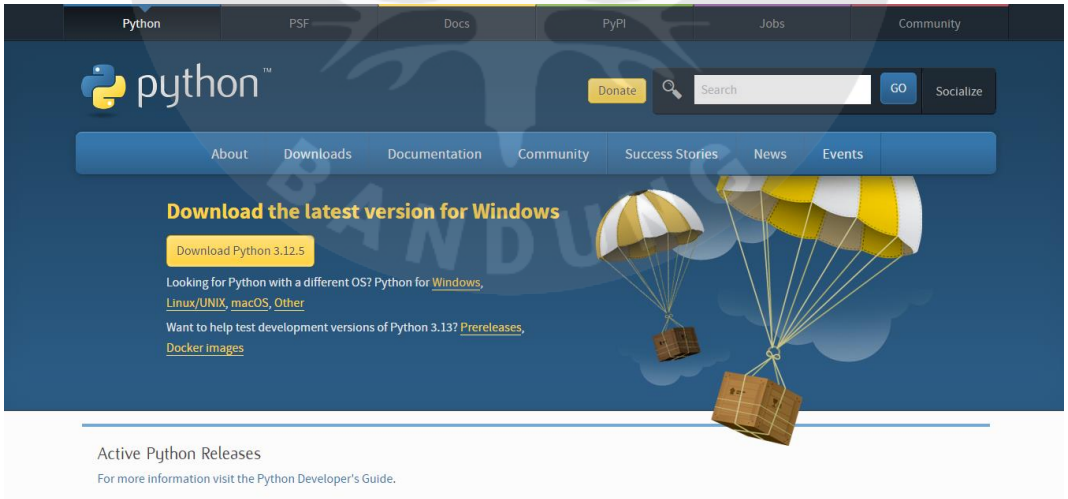
Lampiran 5 Data Hasil Analisis Regresi Benang Rayon Ne_1 30

Regression Statistics	
Multiple R	0,942985098
R Square	0,889220895
Adjusted R Square	0,884404412
Standard Error	0,000110659
Observations	25

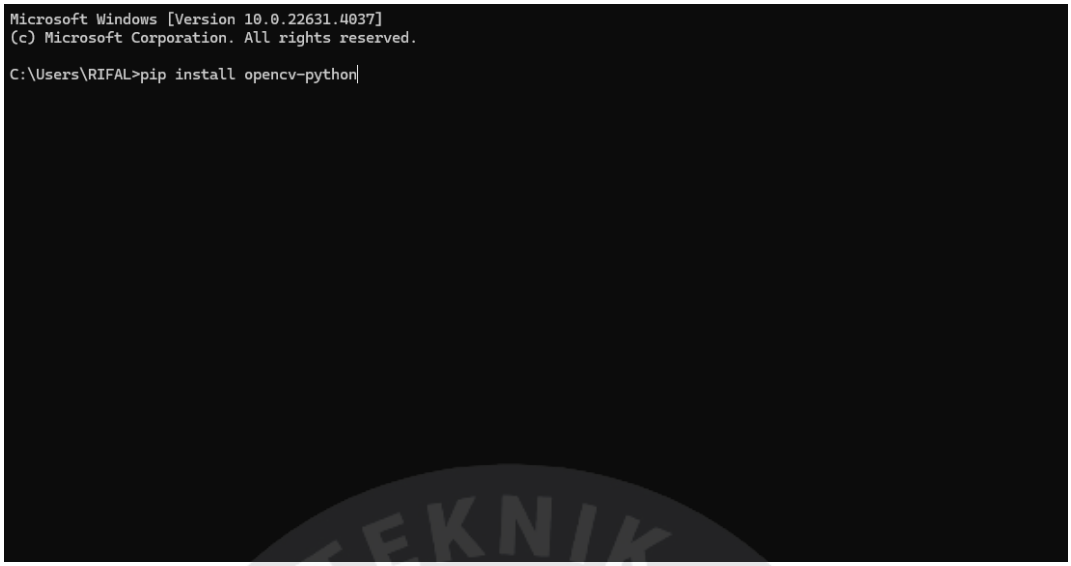
Lampiran 6 Data Hasil Analisis Regresi Benang Rayon Ne_1 30/2

Regression Statistics	
Multiple R	0,915157853
R Square	0,837513895
Adjusted R Square	0,830449282
Standard Error	0,000946036
Observations	25

Lampiran 7 Instalasi *Python*



Lampiran 8 Instalasi Opencv



Lampiran 9 Kode Program

```
import cv2
import numpy as np
# Input Gambar sesuai dengan nama filenya.
image_path = 'Masukan Nama File Gambar.jpg'
calibration_value = Masukkan nilai hasil Kalibrasi  # Nilai kalibrasi dalam mm per piksel
def measure_largest_contour_diameter(image_path, calibration_value):
    # Memuat Gambar
    image = cv2.imread(image_path)
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)  # Konversi ke grayscale
    # Mengurangi Noise untuk mengurangi hairriness
    blurred = cv2.GaussianBlur(gray, (11,5), 9)
    # Menggunakan thresholding global untuk membuat gambar biner
    _, binary = cv2.threshold(blurred, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
    # Mencari Kontur
    contours, _ = cv2.findContours(binary, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
```

```
    if not contours:
        print("Tidak ada kontur ditemukan.")
        return
    # Menghitung panjang (height) bounding box dari setiap kontur dan
menyimpan dalam daftar
    heights = []
    for contour in contours:
        x, y, width, height = cv2.boundingRect(contour)
        heights.append(height)
    # Menemukan panjang kontur terbesar
    height = max(heights)
    # Menghitung diameter benang dalam mm
    diameter_mm = height * calibration_value
    # Mengembalikan diameter benang dalam milimeter dengan 4 angka
desimal
    return diameter_mm
diameter_mm = measure_largest_contour_diameter(image_path,
calibration_value)
print(f"Diameter Benang = {diameter_mm:.4f} mm")
```