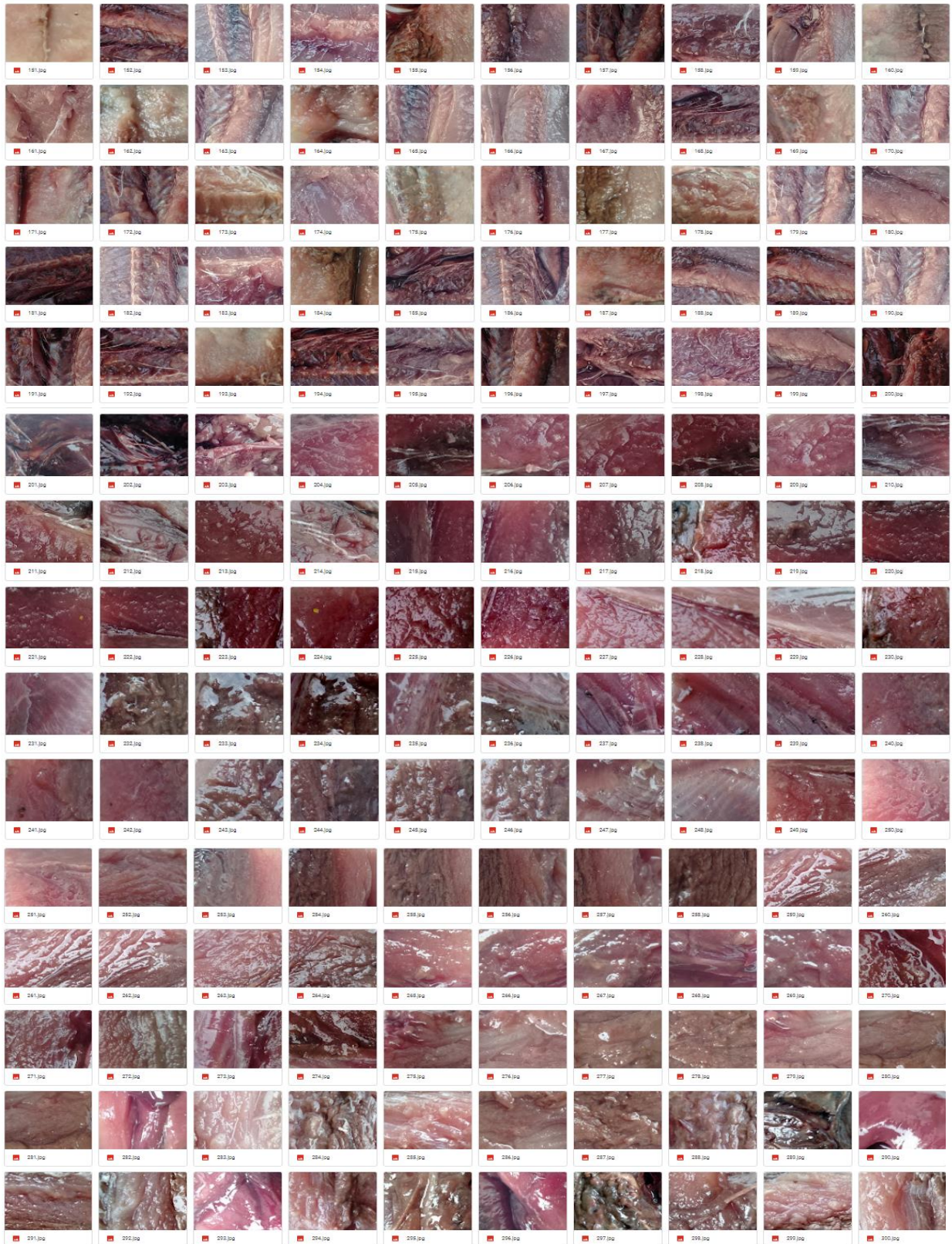


LAMPIRAN

Lampiran 1 Dataset



Lampiran 1 lanjutan *Dataset*



Lampiran 1 lanjutan *Dataset*



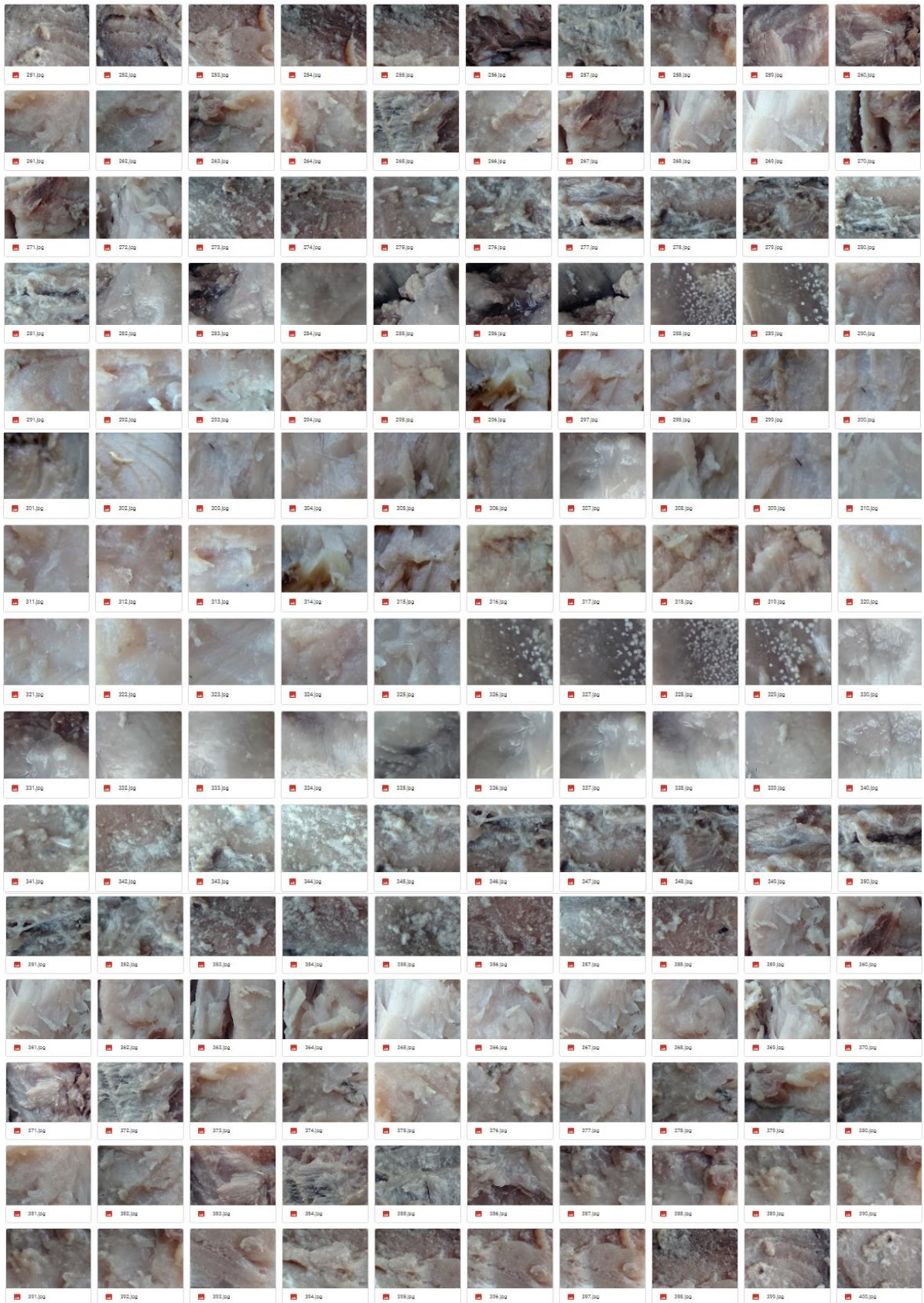
Lampiran 1 Lanjutan *Dataset*



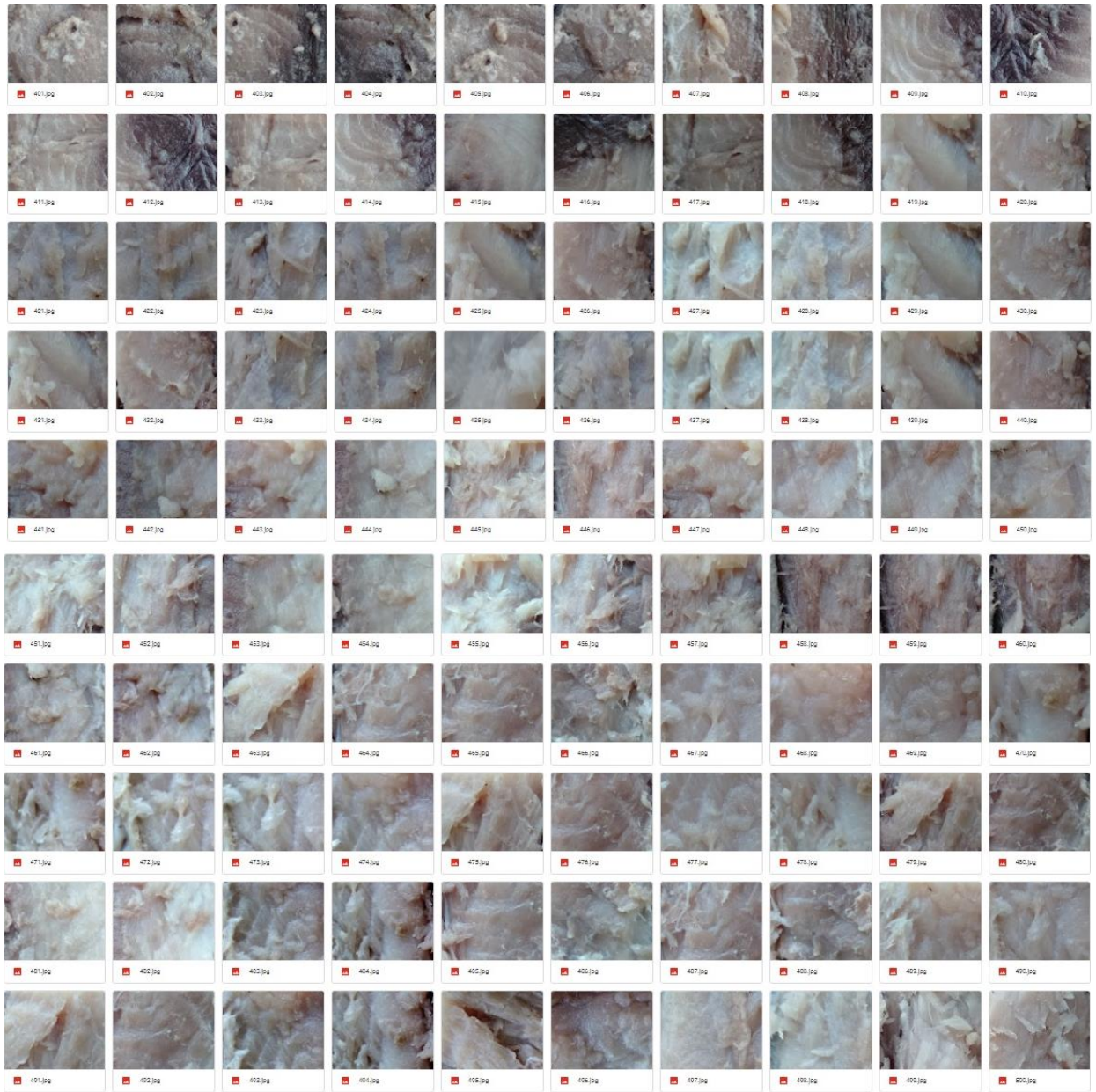
Lampiran 1 Lanjutan *Dataset*



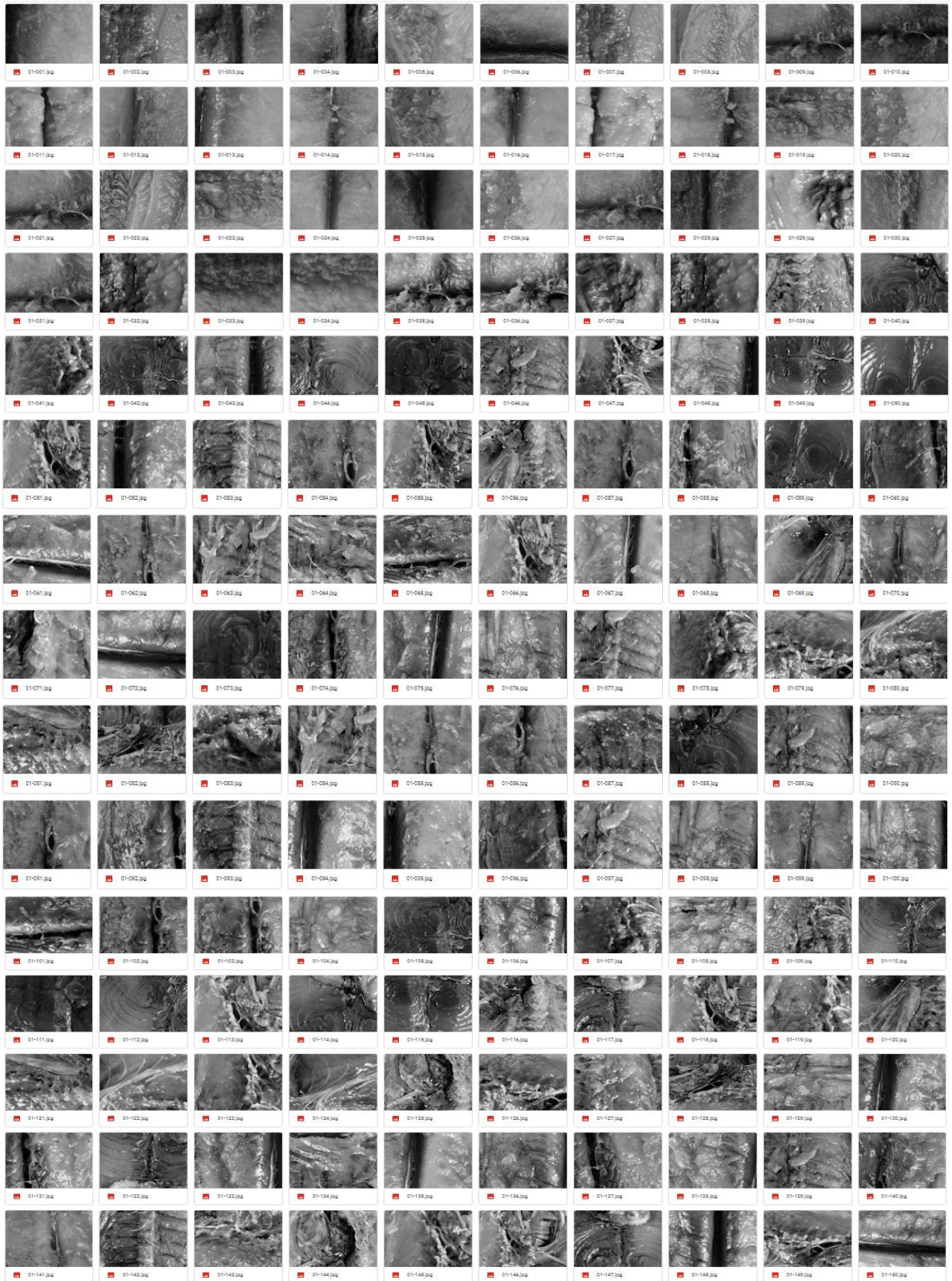
Lampiran 1 Lanjutan *Dataset*



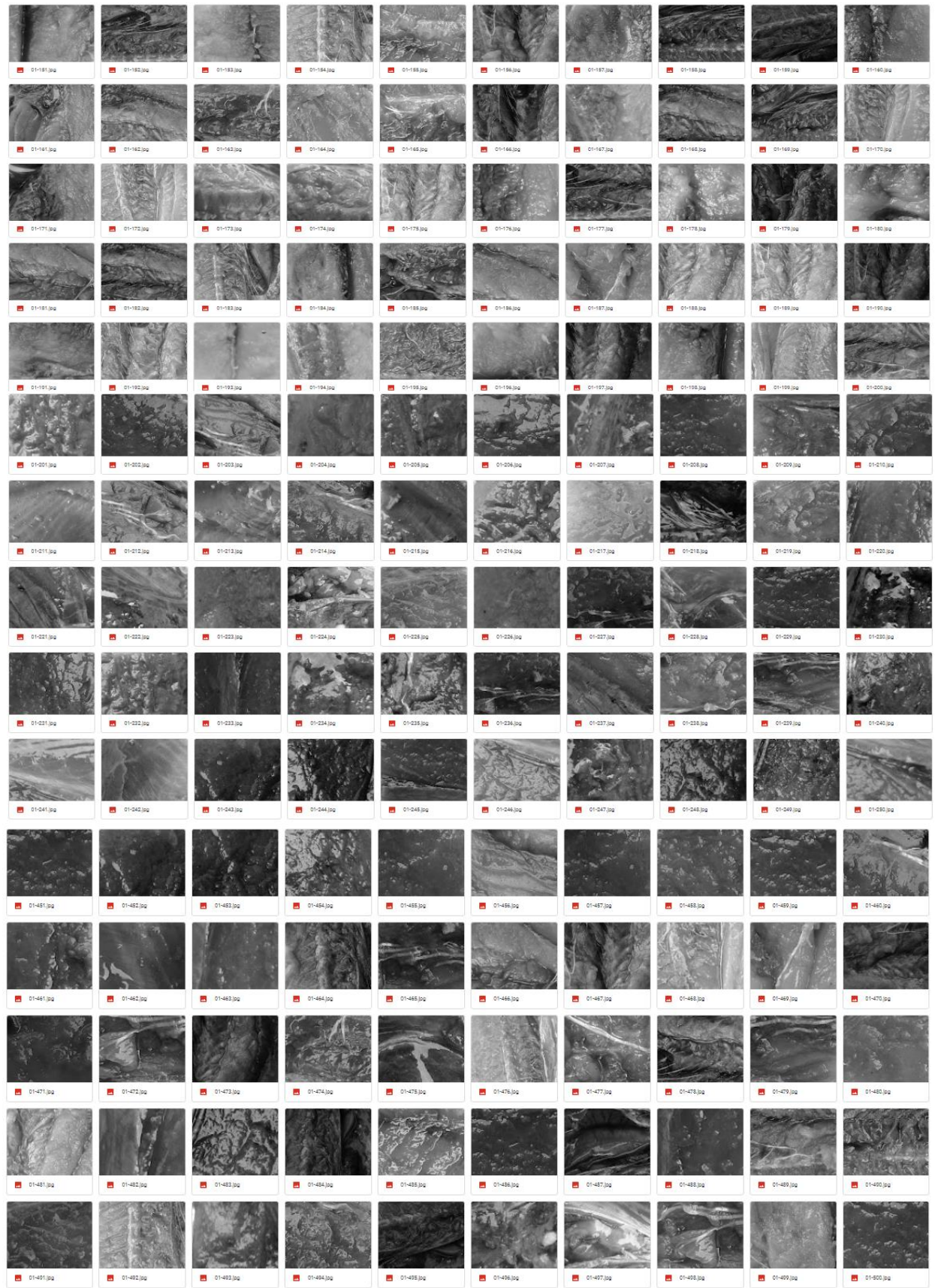
Lampiran 1 Lanjutan *Dataset*



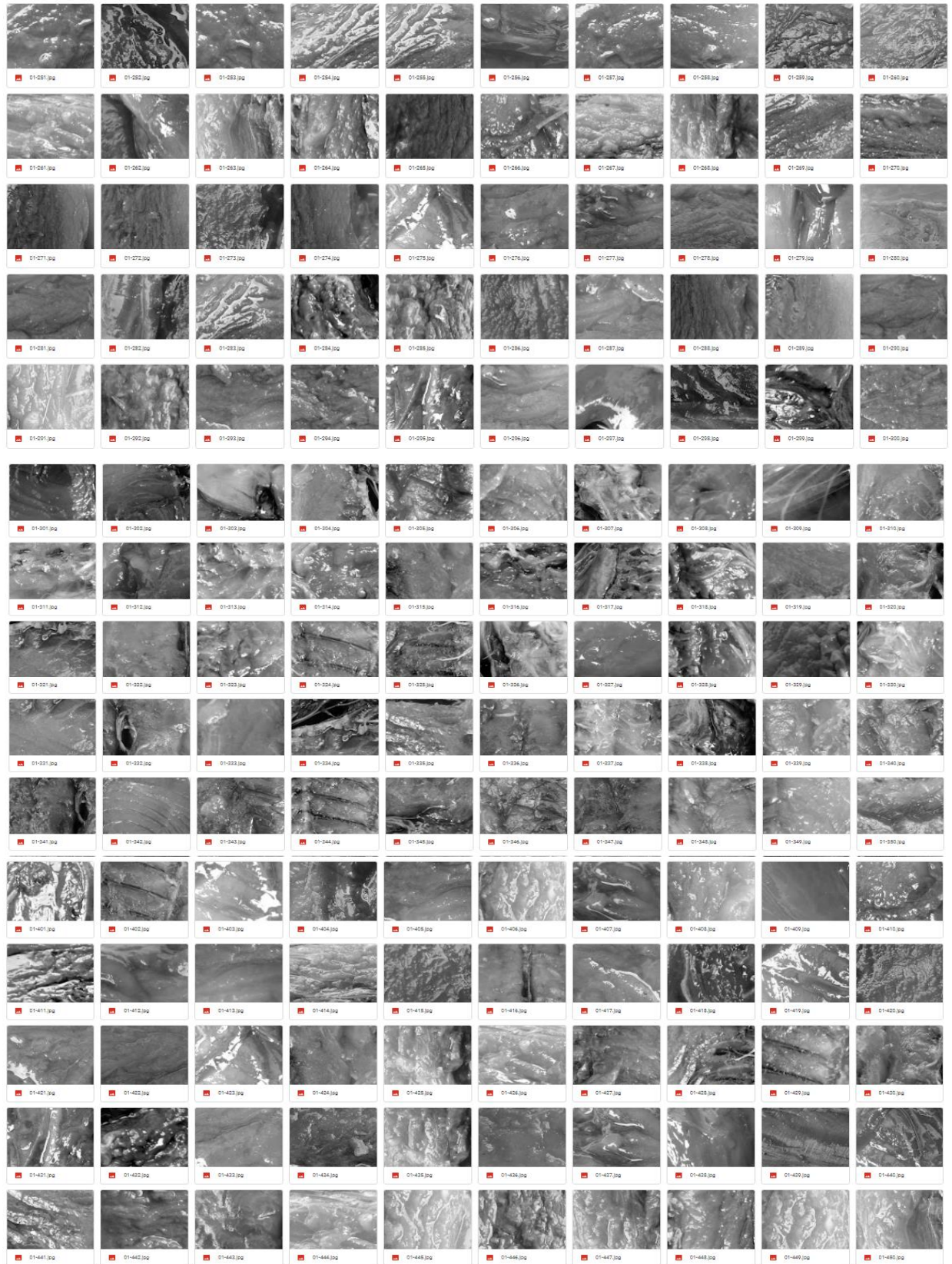
Lampiran 2 Hasil Data Grayscale



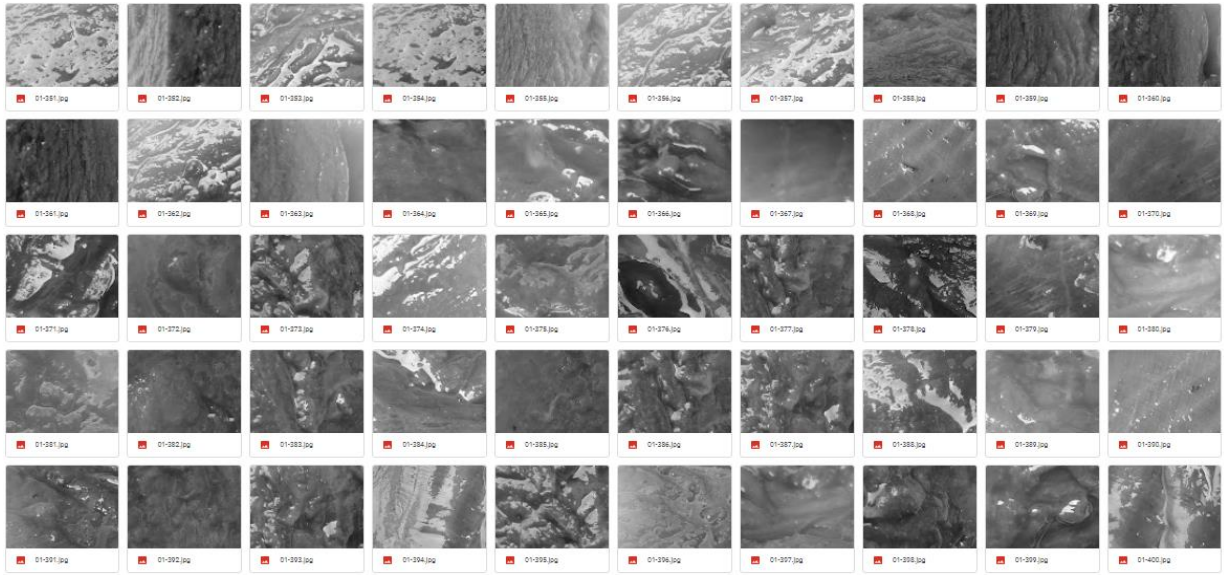
Lampiran 2 lanjutan Hasil Data *Grayscale*



Lampiran 2 lanjutan Hasil Data *Grayscale*



Lampiran 2 lanjutan Hasil Data *Grayscale*



Lampiran 3 Hasil Prediksi dari 200 Data Testing

| Class | Prediksi | | Nama folder |
|-------|----------|-----------|-------------------|
| 544 | 1 | Incorrect | Formalin |
| 515 | 1 | Incorrect | Formalin |
| 193 | 0 | Incorrect | Tidak berformalin |
| 11 | 0 | Incorrect | Tidak berformalin |
| 279 | 0 | Correct | Tidak berformalin |
| 653 | 1 | Incorrect | Formalin |
| 643 | 1 | Correct | Formalin |
| 763 | 1 | Correct | Formalin |
| 198 | 0 | Correct | Tidak berformalin |
| 721 | 1 | Correct | Formalin |
| 236 | 0 | Correct | Tidak berformalin |
| 500 | 1 | Incorrect | Formalin |
| 567 | 1 | Incorrect | Formalin |
| 979 | 1 | Correct | Formalin |
| 969 | 1 | Correct | Formalin |
| 906 | 1 | Correct | Formalin |
| 60 | 0 | Correct | Tidak berformalin |
| 353 | 0 | Incorrect | Tidak berformalin |
| 929 | 1 | Incorrect | Formalin |
| 336 | 0 | Incorrect | Tidak berformalin |
| 817 | 1 | Correct | Formalin |
| 283 | 0 | Correct | Tidak berformalin |
| 948 | 1 | Correct | Formalin |
| 595 | 1 | Incorrect | Formalin |
| 737 | 1 | Correct | Formalin |
| 947 | 1 | Correct | Formalin |
| 3 | 0 | Incorrect | Tidak berformalin |
| 756 | 1 | Correct | Formalin |
| 573 | 1 | Correct | Formalin |
| 944 | 1 | Correct | Formalin |
| 752 | 1 | Correct | Formalin |
| 288 | 0 | Correct | Tidak berformalin |
| 159 | 0 | Incorrect | Tidak berformalin |
| 611 | 1 | Correct | Formalin |
| 347 | 0 | Incorrect | Tidak berformalin |
| 847 | 1 | Correct | Formalin |
| 501 | 1 | Incorrect | Formalin |
| 331 | 0 | Correct | Tidak berformalin |
| 819 | 1 | Correct | Formalin |

Lampiran 3 . Lanjutan Hasil Prediksi dari 200 Data Testing

| Class | | Prediksi | Nama folder |
|-------|---|-----------|-------------------|
| 361 | 0 | Incorrect | Tidak berformalin |
| 470 | 0 | Incorrect | Tidak berformalin |
| 424 | 0 | Incorrect | Tidak berformalin |
| 864 | 1 | Incorrect | Formalin |
| 682 | 1 | Incorrect | Formalin |
| 909 | 1 | Correct | Formalin |
| 72 | 0 | Correct | Tidak berformalin |
| 813 | 1 | Incorrect | Formalin |
| 961 | 1 | Correct | Formalin |
| 629 | 1 | Incorrect | Formalin |
| 316 | 0 | Incorrect | Tidak berformalin |
| 23 | 0 | Incorrect | Tidak berformalin |
| 794 | 1 | Correct | Formalin |
| 286 | 0 | Correct | Tidak berformalin |
| 212 | 0 | Correct | Tidak berformalin |
| 759 | 1 | Incorrect | Formalin |
| 833 | 1 | Correct | Formalin |
| 852 | 1 | Correct | Formalin |
| 530 | 1 | Correct | Formalin |
| 565 | 1 | Incorrect | Formalin |
| 889 | 1 | Correct | Formalin |
| 33 | 0 | Correct | Tidak berformalin |
| 447 | 0 | Correct | Tidak berformalin |
| 555 | 1 | Incorrect | Formalin |
| 664 | 1 | Correct | Formalin |
| 803 | 1 | Incorrect | Formalin |
| 208 | 0 | Correct | Tidak berformalin |
| 647 | 1 | Incorrect | Formalin |
| 131 | 0 | Correct | Tidak berformalin |
| 526 | 1 | Correct | Formalin |
| 841 | 1 | Correct | Formalin |
| 444 | 0 | Incorrect | Tidak berformalin |
| 814 | 1 | Correct | Formalin |
| 593 | 1 | Incorrect | Formalin |
| 438 | 0 | Incorrect | Tidak berformalin |
| 622 | 1 | Incorrect | Formalin |
| 972 | 1 | Correct | Formalin |
| 28 | 0 | Incorrect | Tidak berformalin |
| 445 | 0 | Correct | Tidak berformalin |

Lampiran 3 . Lanjutan Hasil Prediksi dari 200 Data Testing

| Class | Prediksi | | Nama folder |
|-------|----------|-----------|-------------------|
| 862 | 1 | Incorrect | Formalin |
| 258 | 0 | Correct | Tidak berformalin |
| 166 | 0 | Incorrect | Tidak berformalin |
| 830 | 1 | Correct | Formalin |
| 993 | 1 | Correct | Formalin |
| 591 | 1 | Incorrect | Formalin |
| 541 | 1 | Incorrect | Formalin |
| 392 | 0 | Correct | Tidak berformalin |
| 553 | 1 | Incorrect | Formalin |
| 12 | 0 | Incorrect | Tidak berformalin |
| 710 | 1 | Correct | Formalin |
| 938 | 1 | Incorrect | Formalin |
| 800 | 1 | Correct | Formalin |
| 195 | 0 | Correct | Tidak berformalin |
| 312 | 0 | Correct | Tidak berformalin |
| 588 | 1 | Incorrect | Formalin |
| 133 | 0 | Correct | Tidak berformalin |
| 964 | 1 | Correct | Formalin |
| 902 | 1 | Correct | Formalin |
| 577 | 1 | Incorrect | Formalin |
| 176 | 0 | Correct | Tidak berformalin |
| 441 | 0 | Incorrect | Tidak berformalin |
| 106 | 0 | Incorrect | Tidak berformalin |
| 537 | 1 | Incorrect | Formalin |
| 201 | 0 | Correct | Tidak berformalin |
| 698 | 1 | Incorrect | Formalin |
| 370 | 0 | Correct | Tidak berformalin |
| 600 | 1 | Incorrect | Formalin |
| 404 | 0 | Correct | Tidak berformalin |
| 207 | 0 | Incorrect | Tidak berformalin |
| 148 | 0 | Correct | Tidak berformalin |
| 366 | 0 | Correct | Tidak berformalin |
| 309 | 0 | Correct | Tidak berformalin |
| 513 | 1 | Incorrect | Formalin |
| 385 | 0 | Correct | Tidak berformalin |
| 426 | 0 | Incorrect | Tidak berformalin |
| 971 | 1 | Correct | Formalin |
| 356 | 0 | Correct | Tidak berformalin |
| 35 | 0 | Correct | Tidak berformalin |

Lampiran 3 . Lanjutan Hasil Prediksi dari 200 Data Testing

| Citra | Prediksi | | Nama folder |
|-------|----------|-----------|-------------------|
| 614 | 1 | Incorrect | Formalin |
| 420 | 0 | Correct | Tidak berformalin |
| 121 | 0 | Correct | Tidak berformalin |
| 557 | 1 | Incorrect | Formalin |
| 360 | 0 | Correct | Tidak berformalin |
| 352 | 0 | Correct | Tidak berformalin |
| 992 | 1 | Incorrect | Formalin |
| 349 | 0 | Correct | Tidak berformalin |
| 299 | 0 | Correct | Tidak berformalin |
| 269 | 0 | Correct | Tidak berformalin |
| 725 | 1 | Correct | Formalin |
| 59 | 0 | Correct | Tidak berformalin |
| 423 | 0 | Incorrect | Tidak berformalin |
| 250 | 0 | Incorrect | Tidak berformalin |
| 485 | 0 | Incorrect | Tidak berformalin |
| 540 | 1 | Incorrect | Formalin |
| 675 | 1 | Incorrect | Formalin |
| 717 | 1 | Incorrect | Formalin |
| 203 | 0 | Incorrect | Tidak berformalin |
| 439 | 0 | Correct | Tidak berformalin |
| 968 | 1 | Correct | Formalin |
| 869 | 1 | Correct | Formalin |
| 97 | 0 | Incorrect | Tidak berformalin |
| 928 | 1 | Correct | Formalin |
| 395 | 0 | Incorrect | Tidak berformalin |
| 578 | 1 | Correct | Formalin |
| 246 | 0 | Correct | Tidak berformalin |
| 635 | 1 | Incorrect | Formalin |
| 472 | 0 | Incorrect | Tidak berformalin |
| 619 | 1 | Incorrect | Formalin |
| 234 | 0 | Correct | Tidak berformalin |
| 716 | 1 | Correct | Formalin |
| 564 | 1 | Incorrect | Formalin |
| 951 | 1 | Incorrect | Formalin |
| 436 | 0 | Correct | Tidak berformalin |
| 580 | 1 | Correct | Formalin |
| 482 | 0 | Correct | Tidak berformalin |
| 104 | 0 | Correct | Tidak berformalin |
| 863 | 1 | Correct | Formalin |

Lampiran 3 . Lanjutan Hasil Prediksi dari 200 Data Testing

| Citra | Prediksi | | Nama folder |
|-------|----------|-----------|-------------------|
| 559 | 1 | Correct | Formalin |
| 126 | 0 | Correct | Tidak berformalin |
| 831 | 1 | Correct | Formalin |
| 512 | 1 | Correct | Formalin |
| 599 | 1 | Incorrect | Formalin |
| 57 | 0 | Correct | Tidak berformalin |
| 805 | 1 | Correct | Formalin |
| 965 | 1 | Incorrect | Formalin |
| 753 | 1 | Correct | Formalin |
| 343 | 0 | Correct | Tidak berformalin |
| 931 | 1 | Correct | Formalin |
| 786 | 1 | Correct | Formalin |
| 267 | 0 | Correct | Tidak berformalin |
| 505 | 1 | Correct | Formalin |
| 300 | 0 | Incorrect | Tidak berformalin |
| 230 | 0 | Correct | Tidak berformalin |
| 723 | 1 | Correct | Formalin |
| 695 | 1 | Incorrect | Formalin |
| 602 | 1 | Incorrect | Formalin |
| 793 | 1 | Correct | Formalin |
| 138 | 0 | Correct | Tidak berformalin |
| 949 | 1 | Correct | Formalin |
| 939 | 1 | Correct | Formalin |
| 524 | 1 | Incorrect | Formalin |
| 767 | 1 | Correct | Formalin |
| 693 | 1 | Incorrect | Formalin |
| 365 | 0 | Correct | Tidak berformalin |
| 879 | 1 | Correct | Formalin |
| 228 | 0 | Correct | Tidak berformalin |
| 469 | 0 | Correct | Tidak berformalin |
| 244 | 0 | Incorrect | Tidak berformalin |
| 620 | 1 | Correct | Formalin |
| 275 | 0 | Incorrect | Tidak berformalin |
| 666 | 1 | Correct | Formalin |
| 117 | 0 | Correct | Tidak berformalin |
| 247 | 0 | Incorrect | Tidak berformalin |
| 322 | 0 | Correct | Tidak berformalin |
| 458 | 0 | Correct | Tidak berformalin |
| 489 | 0 | Incorrect | Tidak berformalin |

Lampiran 3 . Lanjutan Hasil Prediksi dari 200 Data Testing

| Citra | Prediksi | | Nama folder |
|---------------|----------|-----------|-------------------|
| 860 | 1 | Correct | Formalin |
| 379 | 0 | Correct | Tidak berformalin |
| 896 | 1 | Incorrect | Formalin |
| 809 | 1 | Correct | Formalin |
| 165 | 0 | Correct | Tidak berfotmalin |
| Nilai Akurasi | 60% | | |

Lampiran 4. Hasil Nilai Ekstraksi Tekstur Dan Ekstraksi Warna

| Citra | Class | energo | contrast | homogenity | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | energi | contrast | homoge | correlati | nilai G | nilai B | r |
|--------------------------------------|-------|--------|----------|------------|-----------|--------|----------|---------|-----------|--------|----------|--------|-----------|--------|----------|--------|-----------|--------|----------|--------|-----------|----------|----------|-----------|-----------|--------|----------|--------|-----------|--------|----------|--------|-----------|--------|----------|--------|-----------|---------|---------|---|
| foontendriveMjDivesetseftraainng0100 | GI | 0.0734 | 0.3917 | 18.32321 | 0.9302 | 0.0734 | 0.3917 | 18.3131 | 0.9302 | 0.1935 | 0.3989 | 0.0019 | 0.1073 | 0.3918 | 18.3131 | 0.9302 | 0.0734 | 0.3918 | 18.3131 | 0.9302 | 0.8168963 | 1.25E-05 | 0 | 0.1765125 | | | | | | | | | | | | | | | | |

Lampiran 5. Segmen Program grayscale.py

```
# coding grayscale training
#%% Import libraries
import os
import cv2
# Main
root_folder = '/content/drive/My Drive/ikanku/training'
folders = [os.path.join(root_folder, x) for x in ('/content/drive/My Drive/ikanku/training')]
all_images = [img for folder in folders for img in (folder)]
tgt_base_path = "/content/drive/MyDrive/ikan_abu/training"
for cur_path in os.listdir(root_folder):
    src_sub_path = os.path.join(root_folder, cur_path)
    tgt_sub_path = os.path.join(tgt_base_path, cur_path)
    if not os.path.isdir(tgt_sub_path):
        os.mkdir(tgt_sub_path)
    jdx = 0
    for filename in os.listdir(src_sub_path):
        filepath = os.path.join(src_sub_path, filename)
        img = cv2.imread(filepath, cv2.IMREAD_GRAYSCALE)
        target_path = os.path.join(tgt_sub_path, "%s-%03d.jpg" % (cur_path, jdx+1))
        print(target_path)
        cv2.imwrite(target_path, img)
        jdx += 1
```

Lampiran 6 Segmen Program Resize.py

```
repo_url = 'https://github.com/GotG/object_detection_demo_flow'
'
import os
%cd /content
repo_dir_path = os.path.abspath(os.path.join('.', os.path.base
name(repo_url)))
!git clone {repo_url}
%cd {repo_dir_path}
!git pull

!python resize_images.py --raw-
dir '/content/drive/MyDrive/ikan_abu/training/01' --save-
dir '/content/drive/MyDrive/resize/training/01' --ext jpg --
target-size "(700,700)"

!python resize_images.py --raw-
dir '/content/drive/MyDrive/ikan_abu/training/02' --save-
dir '/content/drive/MyDrive/resize/training/02' --ext jpg --
target-size "(700,700)"
```

Lampiran 7 Segmen Program ekstraksi tekstur.py

```
import math
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import numpy as nm

class Glcm:
    def __init__(self):
        self.result = []
    def forOder(self, a, b):
        size = 0
        for i in a[0]:
            size=size+1
        for i in range(len(a)):
            for j in range(len(a[i])-1):
                p=a[i][j]
                q=a[i][j+1]
                b[p][q]=b[p][q]+1
        matriksIterasil=list(map(list, zip(*b)))
        for i in range(len(b)):
            for j in range(len(b)):
                b[i][j]=b[i][j]+matriksIterasil[i][j]
        #     print("Grayscale Dependenc Mariks")
        #     for i in b:
        #         print(i)
        count=0
        for i in b:
            for j in i:
                count=count+j
        #     print(b)
        return self.normalisasi(b, count)

    def forOder45(self, a, b):
        size = 0
        for i in a[0]:
            size=size+1
        for i in range(len(a)-1):
            for j in range(size-1):
                p=a[i][j+1]
                q=a[i+1][j]
                b[p][q]=b[p][q]+1
        matriksIterasil=list(map(list, zip(*b)))
```

Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
        for i in range(len(b)):
            for j in range(len(b)):

                b[i][j]=b[i][j]+matriksIterasi1[i][j]
#         print("Grayscale Dependenc Mariks 45 derajat")
#         for i in b:
#             print(i)
count=0
for i in b:
    for j in i:
        count=count+j
return self.normalisasi(b, count)

def forOder90(self,a, b):
    size = 0
    for i in a[0]:
        size=size+1
    for i in range(len(a)-1):
        for j in range(size):
            p=a[i][j]
            q=a[i+1][j]
            b[p][q]=b[p][q]+1
    matriksIterasi1=list(map(list,zip(*b)))
    for i in range(4):
        for j in range(4):
            b[i][j]=b[i][j]+matriksIterasi1[i][j]
#         print("Grayscale Dependenc Mariks 90 derajat")
#         for i in b:
#             print(i)
count=0
for i in b:
    for j in i:
        count=count+j
#         print()
return self.normalisasi(b, count)

def forOder135(self,a, b):
    size = 0
    for i in a[0]:
        size=size+1
```

Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
for i in range(len(a)-1):
    for j in range(size-1):
        p=a[i][j]
        q=a[i+1][j+1]
        b[p][q]=b[p][q]+1

matriksIterasi1=list(map(list,zip(*b)))
for i in range(len(b)):
    for j in range(len(b)):
        b[i][j]=b[i][j]+matriksIterasi1[i][j]
#     for i in b:
#         print(i)
count=0
for i in b:
    for j in i:
        count=count+j
return self.normalisasi(b, count)

def normalisasi(self,data, counter):
#     print("Matriks coocurence")
    for i in range(len(data)):
        for j in range(len(data)):
            data[i][j]=data[i][j]/counter
    return data

def asmMethod(self, data):
    sume =0
    for i in data:
        for j in i:
            sume = sume + (j*j)
    return sume

def kontrasMethod(self, data):
    result = 0
    for i in range(len(data)-1):
        for j in range(len(data[i])-1):
            result = result+data[i][j]*((i-j)*(i-j))
    return result
```

Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
def idmMethod(self, data):
    result = 0
    for i in range(len(data)-1):
        for j in range(len(data[i])-1):
            kiri = 1/(1+((i-j)**2))
            tot = kiri*data[i][j]
            result = result + tot
    return result

def ent2Method(self, data):
    result = 0;
    for i in range(len(data)-1):
        for j in range(len(data[i])-1):
            if(data[i][j]>0.0):
                log = math.log2(data[i][j])
            else:
                log = 0
            value = (data[i][j]*log)*(-1)
            result = result + value
    #     print("Entropy : ")
    #     print(result)
    return result

def energy(self,asm):
    return math.sqrt(asm)

#     new correlation method
def getMean(self, data):
    result=[]
    for i in range(len(data)-1):
        for j in range(len(data[i])-1):
            result.append(i*data[i][j])
    return sum(result)
def getStd(self, data, ui):
    result = []
    for i in range(len(data)-1):
        for j in range(len(data[i])-1):
            result.append(data[i][j]*((i-ui)*(i-ui)))
    return sum(result)
```

Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
def correlation(self, data):
    result = []
    u = self.getMean(data)
    mean = self.getStd(data, u)
    for i in range(len(data)-1):
        for j in range(len(data[i])-1):
            temp = ((i-u)*(j-u))/mean
            temp2 = data[i][j]*temp
            result.append(temp2)
    return sum(result)

from PIL import Image
import os
from io import BytesIO
import csv
import numpy as nm

class Data:
    def __init__(self, path):
        imageObject = Image.open(path)
        cropped = imageObject.crop((15,15,85,85))
        self.citra = nm.array(cropped)
        self.citra = self.citra.transpose(2,0,1).reshape(3,-1)
        self.matriksIterasi = [[0 for i in range(256)] for j i
n range(256)]

    def insertoCsv(self,data):
        row = data;
        with open('DatasetGLCM.csv', 'a') as csvFile:
            writer = csv.writer(csvFile)
            writer.writerow(row)
        print("succes")
        csvFile.close()

row =["Citra","Class", "energy0", "contras0", "homogeneity0", "
correlation0", "energy45", "contras45", "homogeneity45", "corre
lation45"
```


Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
        , "energy90", "contras90", "homogeneity90", "correlation90"  
        , "energy135", "contras135", "homogeneity135", "correlation135"  
    "  
        , "energy_mean", "contras_mean", "homogeneity_mean", "cor  
relation_mean"]  
with open('DatasetGLCM.csv', 'a') as csvFile:  
    writer = csv.writer(csvFile)  
    writer.writerow(row)  
    print("succes")  
csvFile.close()
```

```
matriks = [  
    [0,1],  
    [1,0],  
]  
matriksIterasi = [  
    [0,0,0,0],  
    [0,0,0,0],  
    [0,0,0,0],  
    [0,0,0,0],  
]  
glcm = Glcm()  
result = glcm.forOder(matriks, matriksIterasi)  
print("normalisasi 0 derajat : ")  
print(result)  
asm = glcm.asmMethod(result)  
energy = glcm.energy(asm)  
print("energy : ")  
print(energy)  
cont = glcm.contrasMethod(result)  
print("Contras : ")  
print(cont)  
idm = glcm.idmMethod(result)  
print("Homogeneity : ")  
print(idm)  
cor = glcm.correlation(result)  
print("Correlation : ")  
print(cor)
```

Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
glcm = Glcm()
data = Data("/content/1.jpg")
norm = glcm.forOder(data.citra, data.matriksIterasi)
cont = glcm.contrasMethod(norm)
cont

def countfeature(data, glcm):
    asm = glcm.asmMethod(data)
    energy = glcm.energy(asm)
    fature.append(energy)
    idm = glcm.idmMethod(data)
    fature.append(idm)
    cont = glcm.contrasMethod(data)
    fature.append(cont)
    corel = glcm.correlation(data)
    fature.append(corel)

import glob
x =0
for filename in glob.glob('/content/drive/MyDrive/resize/train
ing/01/*.jpg'):
    if x <= 500 :
        data = Data(filename)
        glcm = Glcm()
        fature = [filename, "G1"]
        data0 = glcm.forOder(data.citra, data.matriksIterasi)
        countfeature(data0, glcm)
        data45 = glcm.forOder45(data.citra, data.matriksIterasi)
        countfeature(data45, glcm)
        data90 = glcm.forOder90(data.citra, data.matriksIterasi)
        countfeature(data90, glcm)
        data135 = glcm.forOder135(data.citra, data.matriksIterasi)
        countfeature(data135, glcm)
#         count mean of all feature
        for i in range(len(data0)-1):
            for j in range(len(data0[i])-1):
```

Lampiran 7 Lanjutan. Segmen Program ekstraksi tekstur.py

```
        data0[i][j]=(data0[i][j]+data45[i][j]+data90[i
][j]+data135[i][j])/4
        countfature(data0, glcm)
        data.insertoCsv(fature)
        print(x)
        x=x+1
    else:
        break

import glob
x =0
for filename in glob.glob('/content/drive/MyDrive/resize/train
ing/02/*.jpg'):
    if x <= 500 :
        data = Data(filename)
        glcm = Glcm()
        fature = [filename, "G2"]
        data0 = glcm.forOder(data.citra, data.matriksIterasi)
        countfature(data0, glcm)
        data45 = glcm.forOder45(data.citra, data.matriksIteras
i)
        countfature(data45, glcm)
        data90 = glcm.forOder90(data.citra, data.matriksIteras
i)
        countfature(data90, glcm)
        data135 = glcm.forOder135(data.citra, data.matriksIter
asi)
        countfature(data135, glcm)
        # count mean of all feature
        for i in range(len(data0)-1):
            for j in range(len(data0[i])-1):
                data0[i][j]=(data0[i][j]+data45[i][j]+data90[i
][j]+data135[i][j])/4
                countfature(data0, glcm)
                data.insertoCsv(fature)
                print(x)
                x=x+1
    else:
        break
```

Lampiran 8. Segmen Program ekstraksi warna.py

```
import cv2 as cv
import numpy as np
import os
from imageio import imread

root_folder = '/content/drive/MyDrive/ikanku/training'
folders = [os.path.join(root_folder, x) for x in ('/content/drive/My Drive/ikanku/training/01', '/content/drive/My Drive/ikanku/training/02')]
all_images = [img for folder in folders for img in (folder)]

for cur_path in os.listdir(root_folder):
    src_sub_path = os.path.join(root_folder, cur_path)
    idx = 0
    for filename in os.listdir(src_sub_path):
        filepath = os.path.join(src_sub_path, filename)
        image = cv.imread(filepath, cv.COLOR_RGB2BGR)
        rows, cols, _ = image.shape

        color_B = 0
        color_G = 0
        color_R = 0
        color_N = 0 # neutral/gray color

        for i in range(rows):
            for j in range(cols):
                k = image[i,j]
                if k[0] > k[1] and k[0] > k[2]:
                    color_B = color_B + 1
                    continue
                if k[1] > k[0] and k[1] > k[2]:
                    color_G = color_G + 1
                    continue
                if k[2] > k[0] and k[2] > k[1]:
                    color_R = color_R + 1
                    continue
                color_N = color_N + 1

        pix_total = rows * cols
        print(filepath)
```

Lampiran 8. Lanjutan Segmen Program ekstraksi warna.py

```
        print('Blue:', color_B/pix_total, 'Green:', color_G/pix_total, 'Red:', color_R/pix_total, 'Gray:', color_N/pix_total)
    print("    ")
```

Lampiran 9 Segmen Program KNN.py

```
import numpy as np #array
import matplotlib.pyplot as plt #plot diagram
import pandas as pd #membaca file csv

# Set and input dataset into KNN
dataset = pd.read_csv("/content/abcd500.csv")
y = dataset['Class']
x = dataset.drop('Class',axis = 1)
x = x.drop('Citra', axis = 1)
i =len(x.columns)
x = x.drop(x.columns[i-1], axis = 1)
y.replace(('G1', 'G2',), (0,1), inplace=True)#set class

# split training data and testing data
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x,y,test_s
ize=0.2, random_state=5)

# show the score or akurasi
from sklearn.neighbors import KNeighborsClassifier #library KN
N
clf = KNeighborsClassifier(n_neighbors=1)
clf.fit(X_train, y_train)#training
print(clf.score(X_test, y_test))#testingnya

y_pred = clf.predict(X_test)#prediksi
y_pred

y_test

y_test.to_csv('/content/drive/MyDrive/testknn/testing.csv')
```

Lampiran 9 Lanjutan. Segmen Program KNN.py

```
# predict the testing data
r_test=[]
for i in y_test:

    r_test.append(i)
r_test = np.array(r_test)
print("Class Awal", end= " ; ")
print("Class Prediksi KNN")
for j in range(len(r_test)-1):
    print(r_test[j], end = " ; ")
    print(y_pred[j], end = " => ")
    if(r_test[j]==y_pred[j]):
        print("Correct")
    else:
        print("Incorrect")

class Result:
    def __init__(self, error, akurasi):
        self.error=error
        self.akurasi=akurasi

evaluation=[]
error=[]
# Calculating error for K values between 1 and 40
for i in range(1, 40):
    knn = KNeighborsClassifier(n_neighbors= i)
    knn.fit(X_train, y_train)
    pred_i = knn.predict(X_test)
    error.append(np.mean(pred_i != y_test))
    akurasi=knn.score(X_test, y_test)
    res = Result(np.mean(pred_i != y_test), akurasi)
    evaluation.append(res)

print("Error, Akurasi")
for i in evaluation:
    print(i.error, end = ", ")
    print(i.akurasi)
```

Lampiran 9 Lanjutan. Segmen Program KNN.py

```
plt.figure(figsize=(12, 6))
plt.plot(range(1, 40), error, color='red', linestyle='dashed',
         marker='o',
         markerfacecolor='blue', markersize=10)
plt.title('Error Rate K Value')
plt.xlabel('K Value')
plt.ylabel('Mean Error')
```