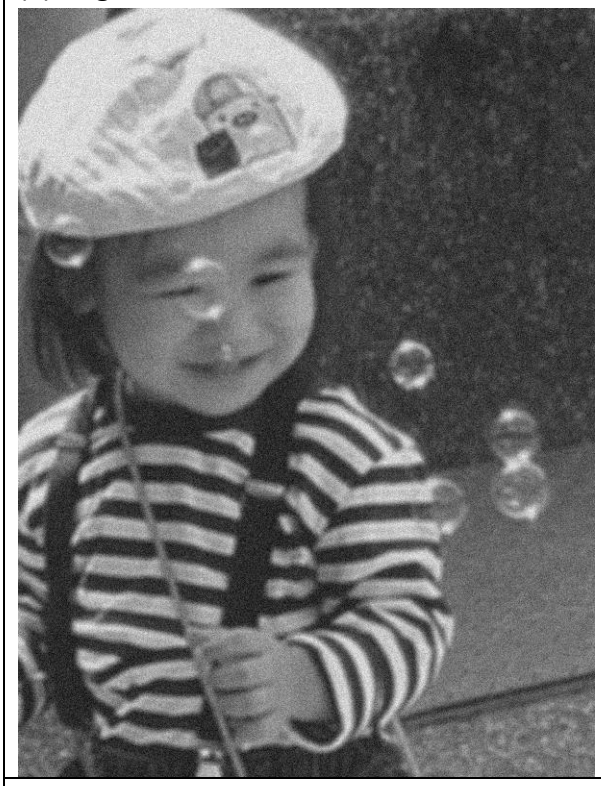
Project1

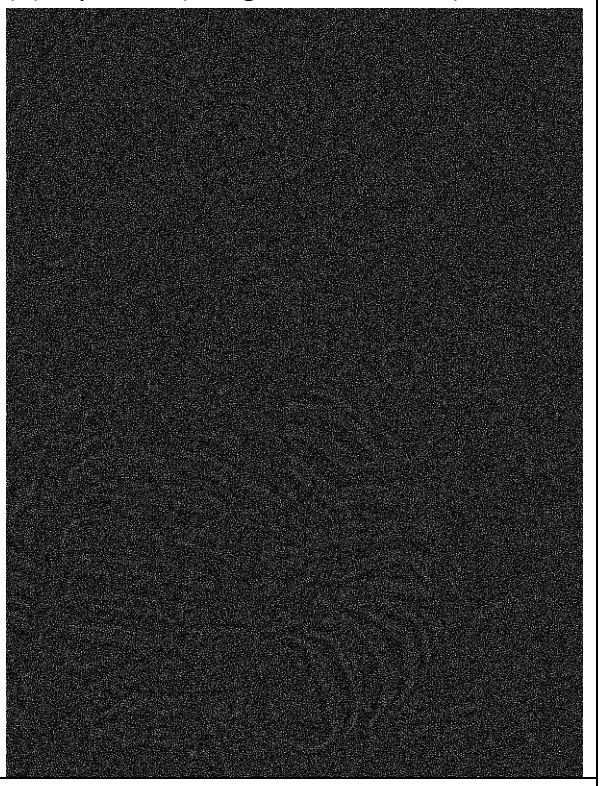
0811562 何祁恩

1. Kid blurred-noisy.tiff

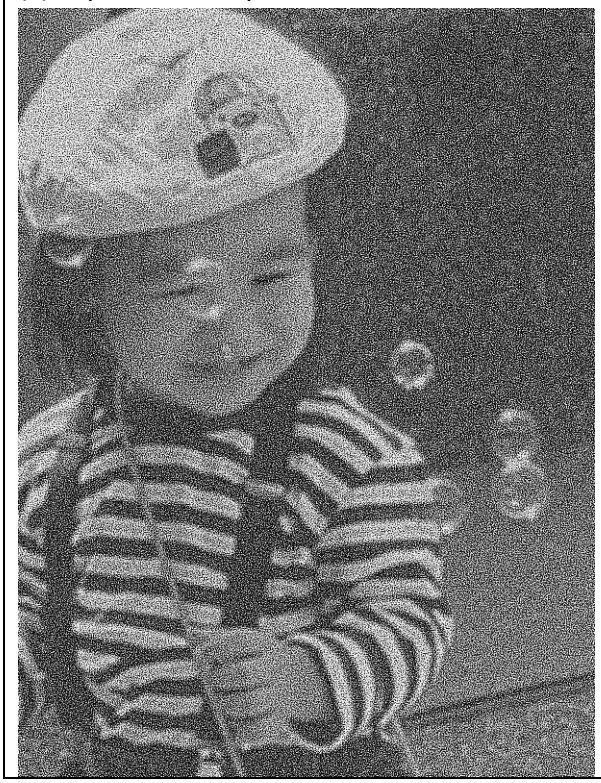
(a) original



(b) Laplacian (using absolute value)



(c) Laplacian-sharpened

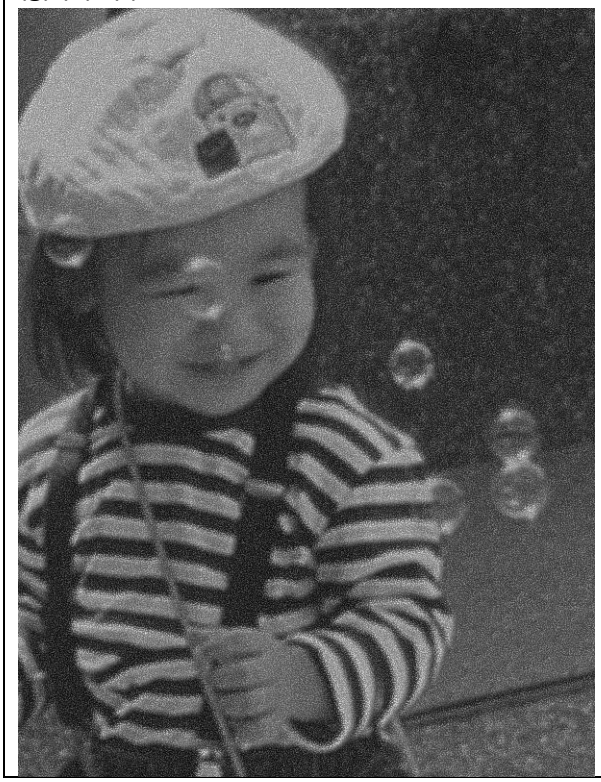


(d) Sobel-gradient

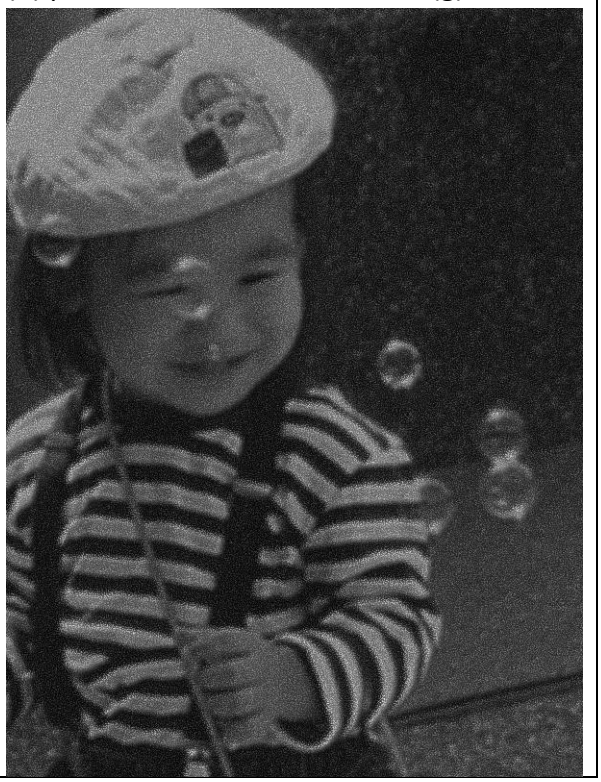
(e) smoothed gradient

(f) (e)・(b)

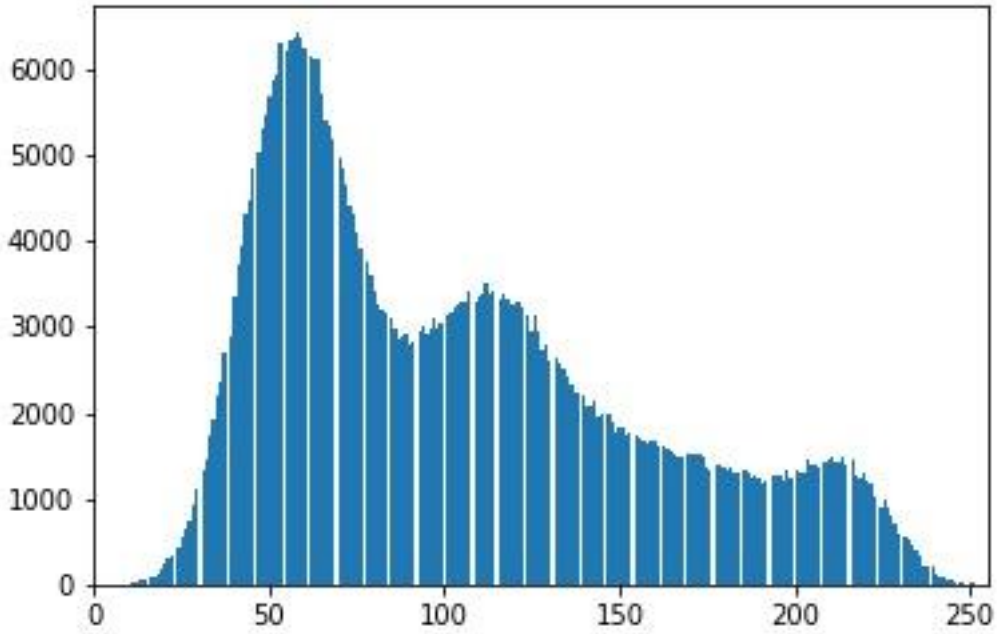
(g) (a)+(f)



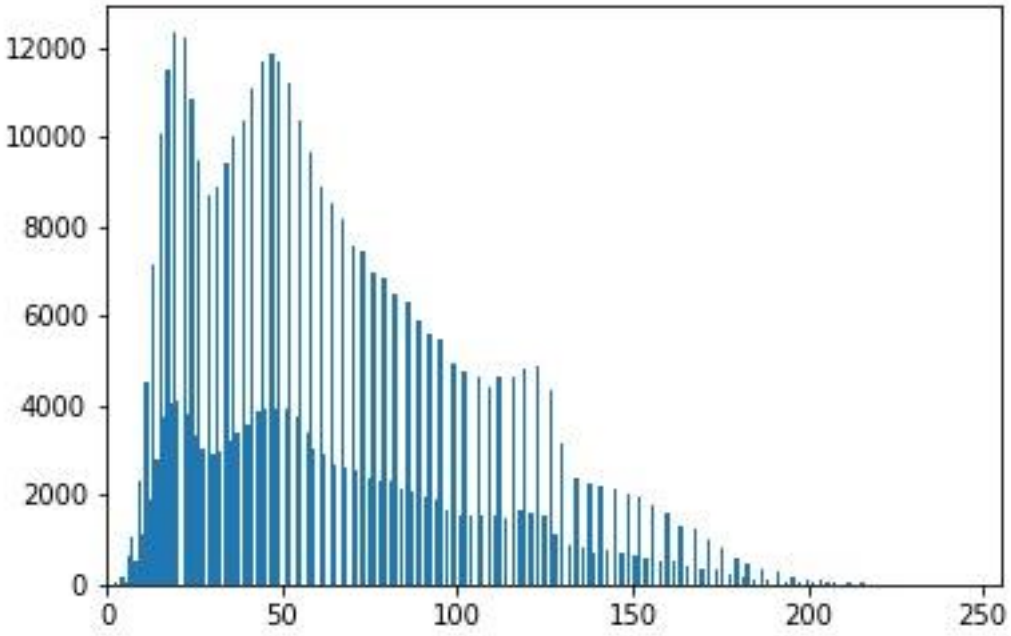
(h) power-law transformation of (g)



Original histograms



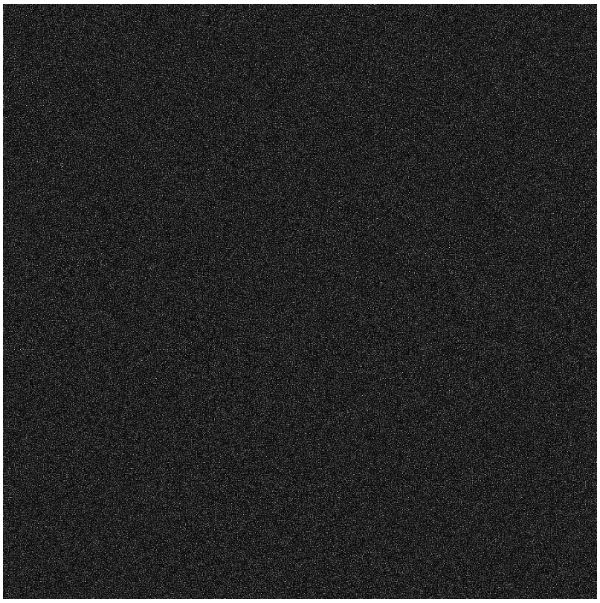
Output histograms



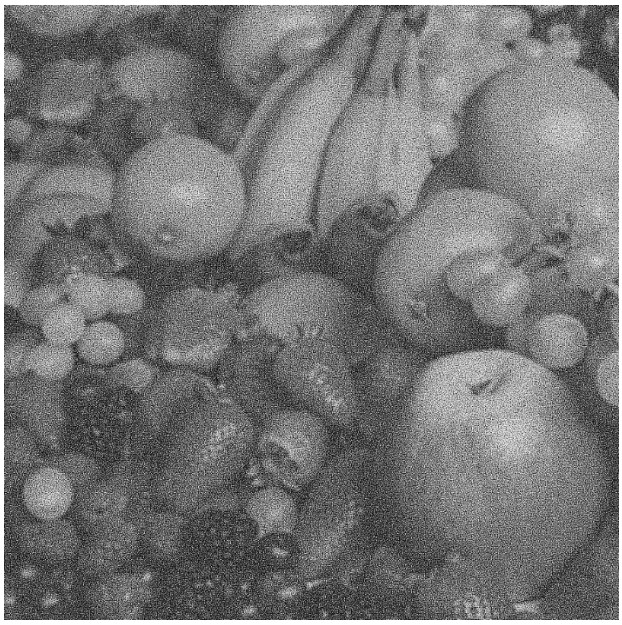
2. Fruit blurred-noisy.tiff

(a) original

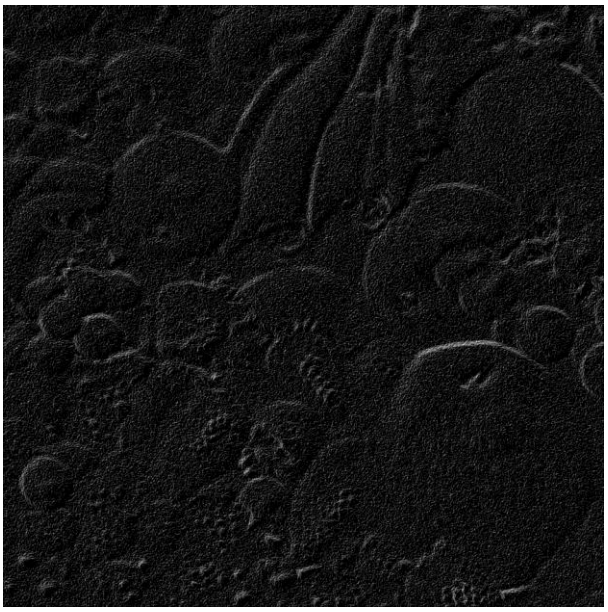
(b) Laplacian (using absolute value)



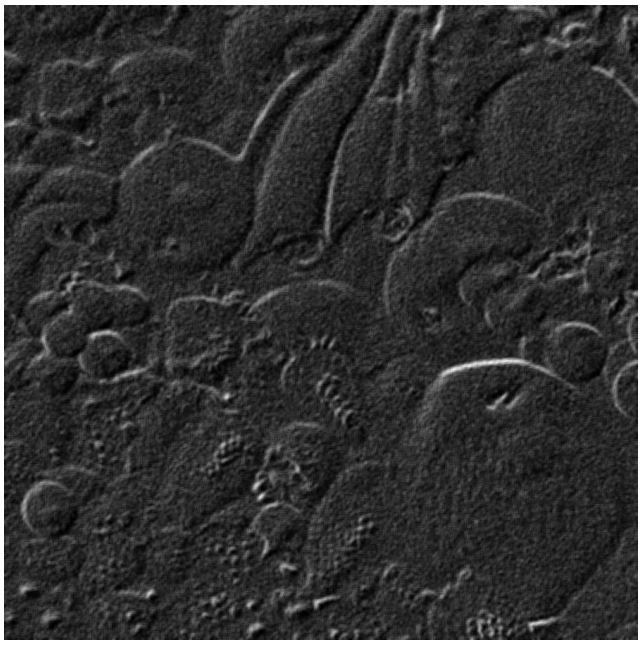
(c) Laplacian-sharpened



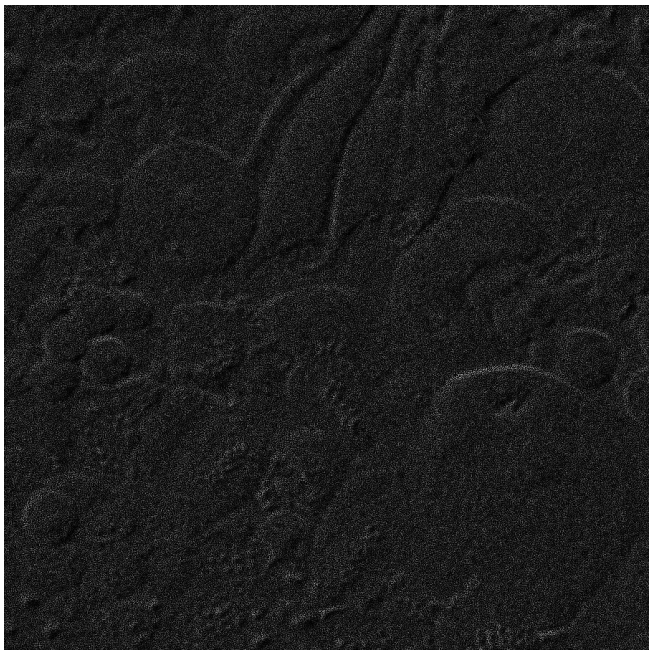
(d) Sobel-gradient



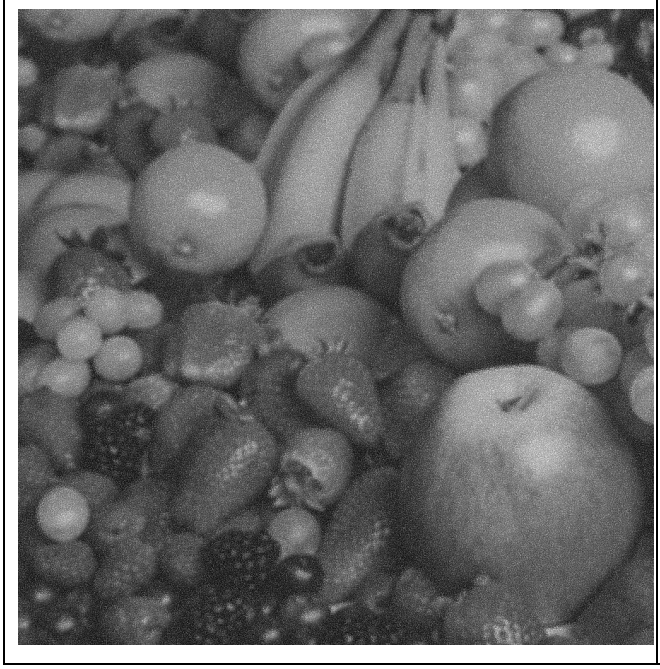
(e) smoothed gradient



(f) (e)・(b)

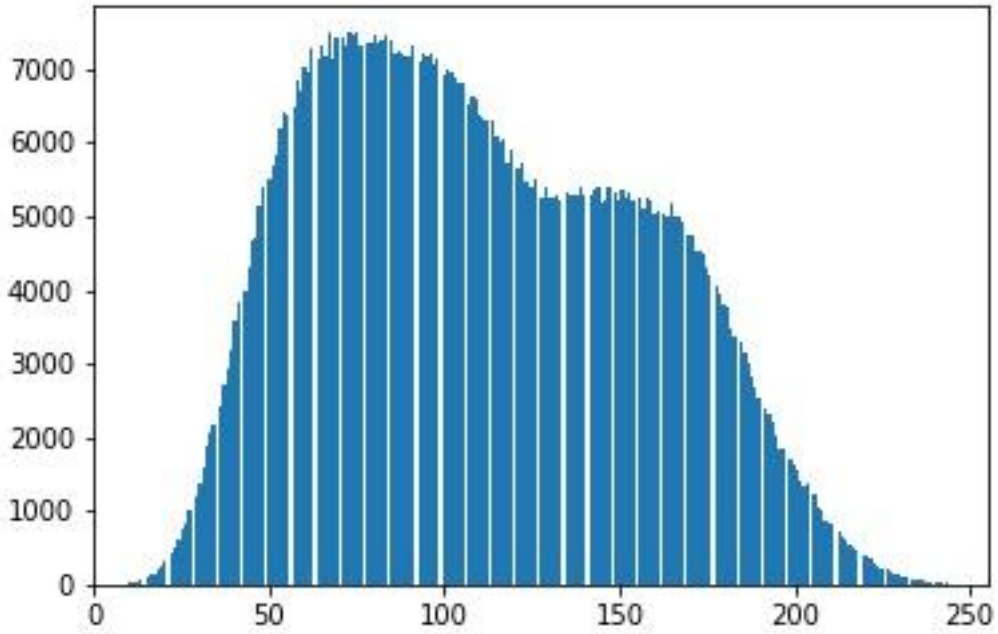


(g) (a)+(f)

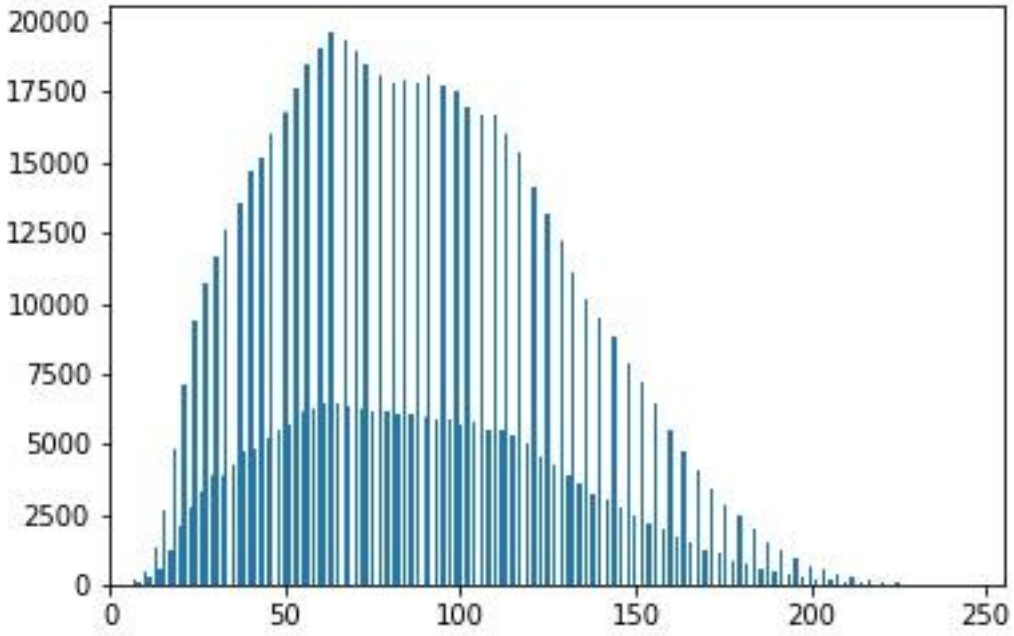


(h) power-law transformation of (g)

Original histograms



Output histograms



SS	*p*(*r*) (kid original)	*p*(*r*) (kid output)	*p*(*r*) (fruit original)	*p*(*r*) (fruit output)
0	2.08E-06	2.08E-06	2.47E-06	2.47E-06
1	0	1.25E-05	3.70E-06	3.70E-06
2	0	7.92E-05	1.23E-06	1.11E-05
3	2.08E-06	3.96E-05	2.47E-06	9.88E-06
4	2.08E-06	0.000314583	7.41E-06	4.32E-05
5	8.33E-06	0.000177083	9.88E-06	0
6	1.25E-05	0.001379167	1.48E-05	3.70E-05
7	0	0.002266667	0	0.000212346
8	1.04E-05	0.001160417	1.48E-05	0.000138272
9	2.71E-05	0.004860417	1.36E-05	0
10	2.92E-05	0.002354167	3.70E-05	0.000632099
11	3.96E-05	0.009377083	5.56E-05	0.00032716
12	6.67E-05	0.003997917	5.68E-05	0
13	0.0001125	0.014875	0.0001	0.001580247
14	0.000135417	0.005777083	0	0.000753086
15	0	0.021008333	0.000138272	0.003245679
16	0.000177083	0.007804167	0.00017037	0
17	0.000222917	0.0239625	0.000192593	0.001490123
18	0.000258333	0.00845	0.000269136	0.005983951
19	0.000372917	0.025708333	0.00032716	0
20	0.000525	0.008552083	0.000404938	0.002524691
21	0.000614583	0	0	0.008741975
22	0.000727083	0.025504167	0.000558025	0
23	0	0.0079125	0.000617284	0.003385185
24	0.000925	0.022579167	0.000753086	0.011576543
25	0.001160417	0.006947917	0.000935802	0
26	0.001358333	0.019727083	0.001049383	0.004106173
27	0.00158125	0.006291667	0.001260494	0.013206173
28	0.001920833	0	0	0
29	0.002354167	0.018164583	0.001491358	0.004779012
30	0	0.006029167	0.001712346	0.014446914
31	0.00275	0.01855	0.001967901	0
32	0.0030375	0.006233333	0.002319753	0.004855556
33	0.00360625	0	0.002544444	0.015511111
34	0.0040125	0.0196375	0.002698765	0
35	0.00459375	0.006689583	0	0.005214815
36	0.004904167	0.020795833	0.003008642	0
37	0.005625	0.007108333	0.003337037	0.016795062

38	0	0	0.003608642	0.005809877
39	0.0060125	0.021545833	0.003925926	0
40	0.006966667	0.0074625	0.004444444	0.01815679
41	0.007720833	0.023060417	0.004744444	0.005985185
42	0.00819375	0	0	0
43	0.008979167	0.0080875	0.004940741	0.018709877
44	0.009329167	0.024375	0.005340741	0
45	0.010079167	0.0081375	0.005758025	0.006451852
46	0	0	0.005816049	0.019804938
47	0.01044375	0.024747917	0.006338272	0
48	0.01103125	0.008220833	0.006665432	0.006781481
49	0.011347917	0.024391667	0	0
50	0.0118125	0	0.006802469	0.020760494
51	0.012222917	0.00814375	0.00702716	0.007053086
52	0.0123125	0.0233375	0.007209877	0
53	0.013095833	0	0.007654321	0.021798765
54	0	0.007760417	0.007920988	0
55	0.012927083	0.021627083	0.007890123	0.007562963
56	0.013170833	0	0	0.02287284
57	0.01323125	0.007039583	0.008025926	0
58	0.013379167	0.020091667	0.00845679	0.007724691
59	0.013247917	0.006347917	0.008277778	0
60	0.01296875	0	0.008687654	0.023553086
61	0	0.0185	0.008598765	0.008004938
62	0.012777083	0.006114583	0.008971605	0
63	0.01275625	0	0	0.024238272
64	0.012733333	0.01770625	0.008828395	0
65	0.01189375	0.005552083	0.009049383	0.008
66	0.011272917	0	0.008851852	0
67	0.01109375	0.016979167	0.009239506	0.023835802
68	0.010783333	0.005410417	0.008823457	0.007854321
69	0	0	0.009165432	0
70	0.010308333	0.015791667	0	0.023377778
71	0.010091667	0.005320833	0.009164198	0
72	0.009714583	0	0.009028395	0.00768642
73	0.009195833	0.0154875	0.009262963	0.022869136
74	0.00895	0	0.009197531	0
75	0.008545833	0.00500625	0.009254321	0.007581481
76	0.008154167	0.014466667	0.00905679	0
77	0	0	0	0.02237037
78	0.007835417	0.004783333	0.009058025	0

79	0.007516667	0.014233333	0.009059259	0.007564198
80	0.00710625	0	0.00922716	0
81	0.006754167	0.004789583	0.009085185	0.022055556
82	0.006629167	0.01353125	0.009124691	0.007534568
83	0.00658125	0	0.009225926	0
84	0	0.004497917	0	0.022090123
85	0.006466667	0	0.009118519	0
86	0.006208333	0.013202083	0.008895062	0.007493827
87	0.005939583	0.004352083	0.008964198	0
88	0.006010417	0	0.008918519	0.022004938
89	0.006054167	0.012327083	0.008869136	0
90	0.005829167	0	0.008859259	0.007341975
91	0.005891667	0.00405625	0.009019753	0.022307407
92	0	0.011639583	0	0
93	0.006114583	0	0.008745679	0.007297531
94	0.006295833	0.003972917	0.008912346	0
95	0.006066667	0.011391667	0.008839506	0.0219
96	0.0062125	0	0.008911111	0
97	0.006429167	0.00349375	0.008739506	0.0073
98	0.0062	0	0.008791358	0
99	0.006335417	0.01031875	0	0.021695062
100	0	0	0.008555556	0.006980247
101	0.006529167	0.003258333	0.008632099	0
102	0.006620833	0.009904167	0.008606173	0.020974074
103	0.006716667	0	0.0085	0
104	0.006795833	0.003222917	0.008409877	0.007081481
105	0.006829167	0	0.008423457	0
106	0.00684375	0.009704167	0	0.020585185
107	0.007114583	0.00319375	0.008038272	0
108	0	0	0.008181481	0.006835802
109	0.006872917	0.009241667	0.008146914	0
110	0.006989583	0	0.007885185	0.020658025
111	0.007022917	0.003202083	0.007835802	0
112	0.007304167	0.009675	0.007775309	0.006774074
113	0.00703125	0	0	0.019781481
114	0.007079167	0.00315	0.007791358	0
115	0	0	0.00752716	0.006508642
116	0.006922917	0.009627083	0.007412346	0
117	0.00701875	0	0.007487654	0.018966667
118	0.006895833	0.0035125	0.007076543	0
119	0.006775	0.0101125	0.007277778	0.006258025

120	0.006766667	0	0	0
121	0.006841667	0.00335625	0.006961728	0.017432099
122	0.006695833	0	0.007067901	0
123	0	0.01015625	0.00675679	0.005592593
124	0.0065125	0	0.006750617	0
125	0.006135417	0.003210417	0.006661728	0.016254321
126	0.006529167	0	0.006783951	0
127	0.006125	0.00901875	0	0.005249383
128	0.005679167	0.002404167	0.006491358	0
129	0.005804167	0	0.006650617	0.015139506
130	0.005452083	0.006614583	0.006483951	0
131	0	0	0.006493827	0.004746914
132	0.005491667	0.001895833	0.006535802	0.013679012
133	0.00538125	0	0.006459259	0
134	0.005227083	0.005004167	0	0.004412346
135	0.0050625	0	0.006591358	0
136	0.004866667	0.001704167	0.006519753	0.012549383
137	0.004835417	0	0.006511111	0
138	0.004622917	0.004775	0.00654321	0.004016049
139	0	0.001433333	0.006646914	0
140	0.00460625	0	0.006520988	0.011724691
141	0.004358333	0.004641667	0	0
142	0.004335417	0	0.006512346	0.003797531
143	0.004464583	0.001558333	0.006614815	0
144	0.004064583	0	0.006659259	0.01084321
145	0.004158333	0.004475	0.006406173	0
146	0	0	0.006439506	0.003338272
147	0.00414375	0.001439583	0.006653086	0
148	0.00396875	0	0	0.009738272
149	0.00369375	0.004260417	0.006582716	0
150	0.003839583	0	0.006453086	0.00308642
151	0.003833333	0.001320833	0.006630864	0
152	0.003622917	0.0041375	0.006506173	0.008858025
153	0.003675	0	0.006566667	0
154	0	0.001277083	0.006434568	0.002719753
155	0.003635417	0	0	0
156	0.0035375	0.0037375	0.006493827	0.007969136
157	0.003497917	0	0.006325926	0
158	0.0034	0.001179167	0.006475309	0.002423457
159	0.00348125	0	0.006440741	0
160	0.003466667	0.003358333	0.00621358	0.006822222

161	0.003341667	0	0.00625679	0
162	0	0.00109375	0	0.002101235
163	0.00336875	0	0.006212346	0
164	0.003283333	0.002741667	0.006190123	0.005846914
165	0.003260417	0	0.006414815	0
166	0.003189583	0.00091875	0.006179012	0.001818519
167	0.003104167	0	0.006161728	0
168	0.003116667	0.002641667	0.006087654	0.004990123
169	0	0	0	0
170	0.0031375	0.00075	0.005855556	0.001559259
171	0.003177083	0	0.005838272	0
172	0.0031875	0.002127083	0.00557284	0.004233333
173	0.003147917	0	0.005585185	0
174	0.003072917	0.000702083	0.005524691	0.001359259
175	0.002858333	0	0.005317284	0
176	0.002777083	0.00168125	0.005191358	0.003538272
177	0	0	0	0
178	0.002910417	0.0005375	0.005018519	0.001061728
179	0.002891667	0	0.00487284	0
180	0.002866667	0.001302083	0.004701235	0.003039506
181	0.00280625	0	0.004638272	0
182	0.002852083	0.000375	0.004295062	0.000890123
183	0.002704167	0.001054167	0.004138272	0
184	0.002689583	0	0	0.002445679
185	0	0.000295833	0.004049383	0
186	0.00276875	0	0.003902469	0.000680247
187	0.002702083	0.000802083	0.003708642	0
188	0.002608333	0	0.003488889	0.001907407
189	0.002633333	0.000241667	0.003325926	0
190	0.002589583	0	0.003149383	0.000537037
191	0.002475	0	0	0
192	0.002497917	0.0005625	0.002971605	0.001496296
193	0	0	0.002848148	0
194	0.002647917	0.000141667	0.002741975	0.000422222
195	0.002620833	0	0.002525926	0
196	0.002627083	0.0003875	0.002298765	0.0012
197	0.0025375	0	0.002302469	0
198	0.002779167	0.0001	0	0.000318519
199	0.002597917	0	0.00211358	0
200	0	0.000277083	0.00202716	0.000853086
201	0.002747917	0	0.001928395	0

202	0.002729167	8.75E-05	0.00172716	0.000234568
203	0.002739583	0	0.001644444	0
204	0.003039583	0.00020625	0.001683951	0.000675309
205	0.00291875	0	0	0
206	0.002914583	7.92E-05	0.001528395	0.000195062
207	0.002845833	0	0.001293827	0
208	0	0.00013125	0.001250617	0.00045679
209	0.002954167	0	0.001092593	0
210	0.003010417	6.04E-05	0.001025926	0.000123457
211	0.00311875	0	0.001025926	0
212	0.002958333	8.13E-05	0	0.000322222
213	0.003002083	0	0.00088642	0
214	0.0031125	1.04E-05	0.000812346	0
215	0.002879167	0	0.000766667	8.64E-05
216	0	6.25E-05	0.000667901	0
217	0.003052083	0	0.000639506	0.000220988
218	0.002641667	2.08E-05	0.000593827	0
219	0.002570833	0	0	5.19E-05
220	0.002735417	4.79E-05	0.00047284	0
221	0.002502083	0	0.00047284	0.00012716
222	0.00244375	2.08E-06	0.000444444	0
223	0.0021625	0	0.000365432	3.58E-05
224	0	1.46E-05	0.000328395	0
225	0.001891667	0	0.000264198	0.000122222
226	0.002075	0	0	0
227	0.0019	1.04E-05	0.000274074	3.58E-05
228	0.001647917	0	0.000238272	0
229	0.001485417	1.88E-05	0.000183951	5.19E-05
230	0.001214583	0	0.000192593	0
231	0	6.25E-06	0.000124691	1.23E-05
232	0.001147917	0	0.000133333	0
233	0.001072917	1.25E-05	0	3.83E-05
234	0.000977083	0	9.38E-05	0
235	0.000839583	4.17E-06	9.01E-05	4.94E-06
236	0.00073125	0	7.04E-05	0
237	0.000483333	6.25E-06	7.53E-05	0
238	0.000435417	0	5.19E-05	3.33E-05
239	0	0	4.07E-05	0
240	0.000427083	0	0	1.23E-06
241	0.000272917	0	2.84E-05	0
242	0.000166667	4.17E-06	2.72E-05	1.11E-05

243	0.000191667	0	1.73E-05	0
244	0.00015625	0	2.59E-05	3.70E-06
245	0.000114583	0	8.64E-06	0
246	9.38E-05	0	9.88E-06	6.17E-06
247	0	0	0	0
248	7.50E-05	2.08E-06	6.17E-06	3.70E-06
249	2.92E-05	0	2.47E-06	0
250	2.71E-05	2.08E-06	2.47E-06	6.17E-06
251	3.54E-05	0	1.23E-06	0
252	1.46E-05	0	4.94E-06	0
253	1.04E-05	0	1.23E-06	0
254	0	0	0	0
255	4.17E-06	2.08E-06	1.23E-06	1.23E-06

3.code

Import package:

```
import numpy as np
import cv2
from PIL import Image
import matplotlib.pyplot as plt
np.set_printoptions(threshold = np.inf)
```

Function to save picture/constrast_stretching/plot histogram/calculate probability

```
def save_picture(filename, source):
  image = Image.fromarray(np.uint8(source))
  image.save('./'+filename, dpi=(200, 200))

def constrast_stretching(source):
  min = np.min(source)
  max = np.max(source)
  L = 256
  source = (source - min) / (max - min) * (L - 1)
  return source.astype(np.uint8)

def histogram(source, filename):
  plt.hist(source.ravel(),256,[0,255])
  plt.xlim([0,255])
  plt.savefig(filename)
  plt.show()

def cal_probability(source):
    MN = source.shape[0] * source.shape[1]
    ret = []
    source = source.flatten().tolist()
    for i in range(0, 256):
        ret.append(source.count(i)/MN)

    for i in range(0, 256):
        print(ret[i])
```

DIP main function:

```
def DIP(filename, gamma):

    #original
    original = cv2.imread(filename + '.tif', cv2.IMREAD_GRAYSCALE)
    save_picture(filename+'_(a).jpg', constrast_stretching(original))
    print(cal_probability(constrast_stretching(original)))
    histogram(constrast_stretching(original), filename + '_original_histgram.jpg')

    #Laplacian
    kernel = np.array([[-1, -1, -1],
                       [-1,  8, -1],
                       [-1, -1, -1]], dtype = np.double)
    Laplacian = cv2.filter2D(original, ddepth = 16, kernel=kernel)
    save_picture(filename+'_(b).jpg', np.abs(constrast_stretching(Laplacian)))

    #Laplacian sharpen
    kernel = np.array([[-1, -1, -1],
                       [-1,  9, -1],
                       [-1, -1, -1]], dtype = np.double)
    Laplacian_sharpen = cv2.filter2D(original, ddepth = 16, kernel=kernel)
    save_picture(filename+'_(c).jpg', constrast_stretching(Laplacian_sharpen))
```

```
    #Sobel gradient
    kernel = np.array([[-1, -2, -1],
                       [0,   0,  0],
                       [1,   2,  1]], dtype = np.double)
    gx = cv2.filter2D(original, ddepth=16, kernel=kernel)

    kernel = np.array([[-1, 0, 1],
                       [-2, 0, 2],
                       [-1, 0, 1]], dtype = np.double)
    gy = cv2.filter2D(original, ddepth=16, kernel=kernel)

    Sobel_gradient = cv2.addWeighted(np.abs(gx), 0.5, np.abs(gy),0.5,0)
    save_picture(filename+'_(d).jpg', constrast_stretching(Sobel_gradient))

    #smooth gradient
    kernel = np.ones([5, 5]) / 25
    smooth_gradient = cv2.filter2D(Sobel_gradient, ddepth=16, kernel=kernel)
    save_picture(filename+'_(e).jpg', constrast_stretching(smooth_gradient))

    #extracted feature
    extracted_feature = cv2.multiply(np.sqrt(smooth_gradient).astype(np.uint8),np.sqrt(Laplacian).astype(np.uint8))
    save_picture(filename+'_(f).jpg', constrast_stretching(extracted_feature))

    g = cv2.addWeighted(original, 0.5, extracted_feature,0.5,0, dtype = 16)
    save_picture(filename+'_(g).jpg', constrast_stretching(g))

    #power law
    h = np.array(255*(g / 255) ** gamma)
    save_picture(filename+'_(h).jpg', constrast_stretching(h))
    print(cal_probability(constrast_stretching(h)))
    histogram(constrast_stretching(h), filename + '_output_histgram.jpg')
```

Driven code:

```
if __name__ == "__main__":
    DIP('kid blurred-noisy', gamma = 1.4)
    DIP('fruit blurred-noisy', gamma = 1.2)
```