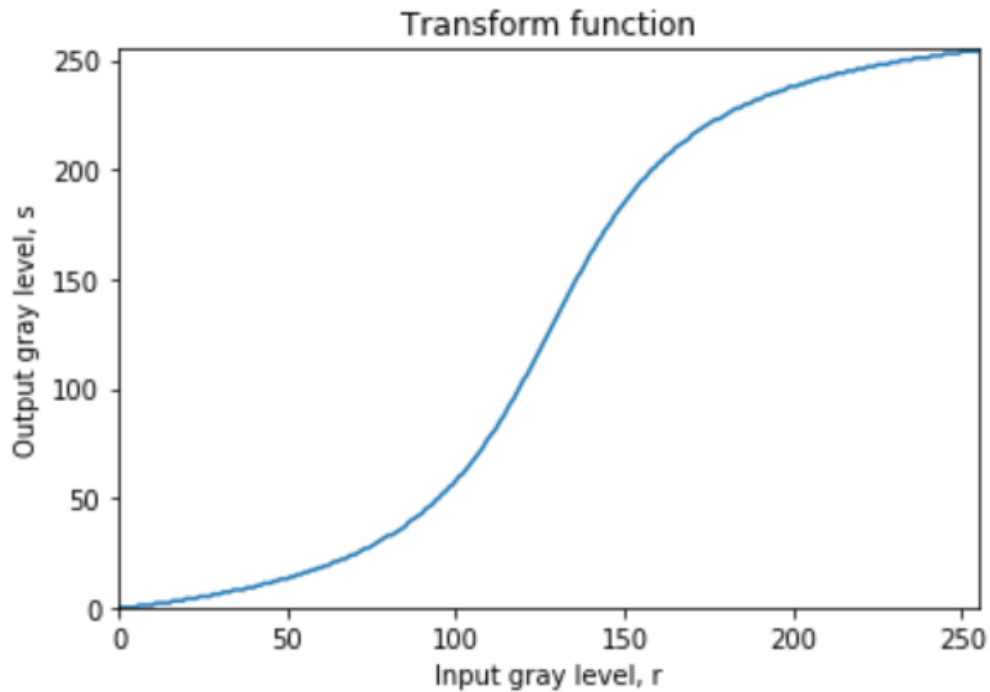


## Project #1

1. The transform function

$$s = T(r) = (\arctan((r - 128.0)/32) + 1.325817) * 96.2343$$

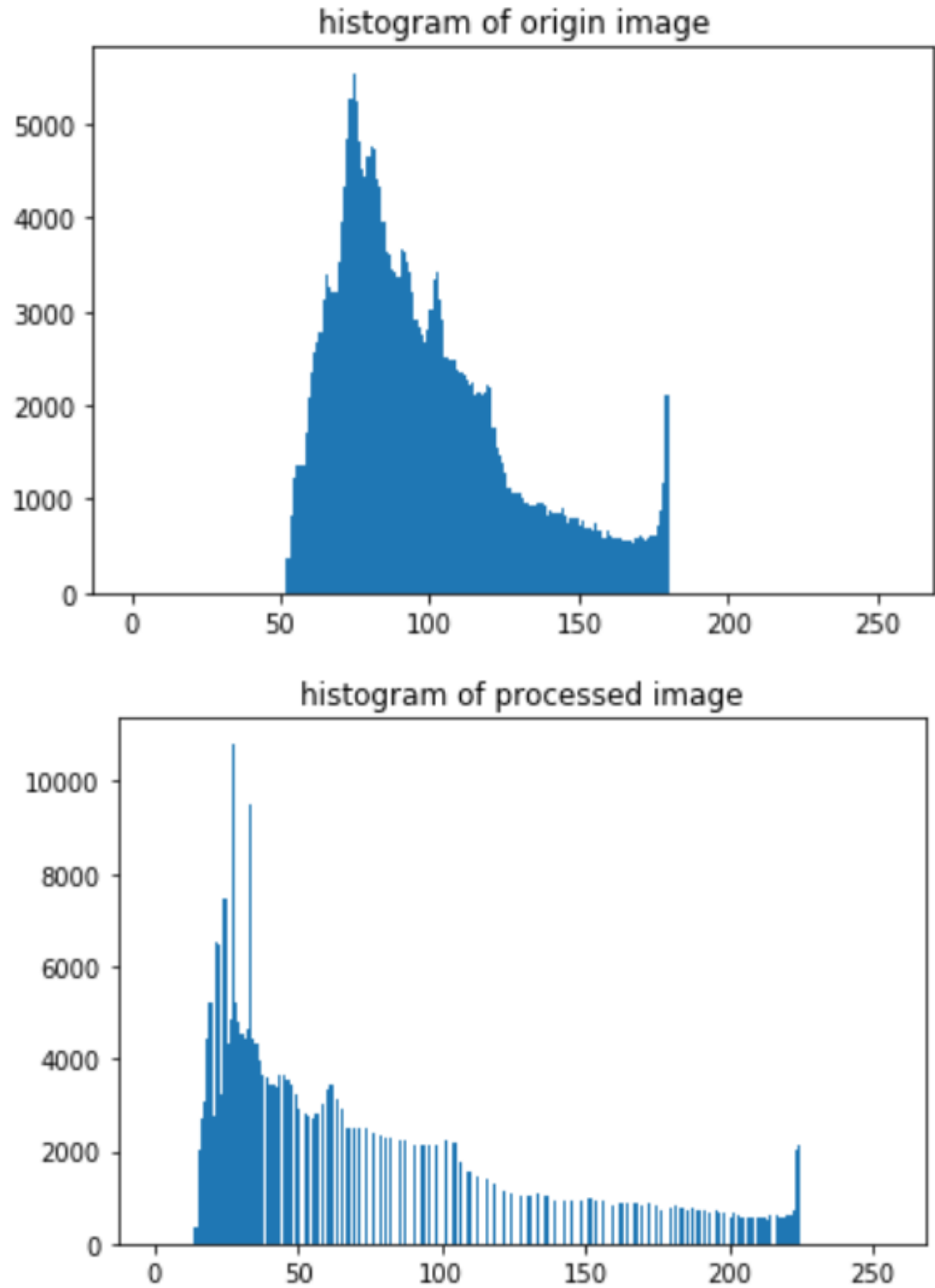
2. Figure of the transform function



3. Figure of the output image after applying the intensity transformation function



4. Figures of the original and output histograms



5. Table of transformation function

	0	1	2	3	4	5	6	7	8	9	...	246	247	248	249	250	251	252	253	254	255
output	0	0	0	0	0	0	1	1	1	1	...	253	253	253	253	254	254	254	254	254	254

上面為 input gray level r.

## 6. Source code

```
# Library
import cv2
import numpy as np
from matplotlib import pyplot as plt
```

```
# read image and transform gray scale
img = cv2.imread('Bird feeding 3 low contrast.tif')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

```
#  $s = T(r)$ 
s = ((np.arctan((gray-128.0)/32)+1.325817)*96.2343).astype(np.uint8)
```

```
# show the processed image
cv2.imshow("test",s)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
# show the histogram of origin image
plt.hist(gray.ravel(),256,[0,256])
plt.title("histogram of origin image")
```

```
# show the histogram of processed image
plt.hist(s.ravel(),256,[0,256])
plt.title("histogram of processed image")
```

```
# show the transform function
x = np.arange(256)
y = ((np.arctan((x-128.0)/32)+1.325817)*96.2343).astype(np.uint8)
plt.plot(x,y)
plt.axis([0,255,0,255])
plt.title("Transform function")
plt.xlabel("Input gray level, r")
plt.ylabel("Output gray level, s")
```

```
import pandas as pd

# combine input and output
dictionary = dict(zip(x, y))
index = ["output"]
```

```
# show this table
table = pd.DataFrame(dictionary,index)
table
```