COMP 4421 (Fall 2018)

Assignment #3

Name: CHEUNG, Daniel Student ID: 20423088

E-mail:dcheungaa@connect.ust.hk

I. Exercises

1) Huffman Coding

x_i	Count
1	4
9	3
3	3
7	2
2	1
12	1
15	1
Total	15

x_i	$P(x_i)$					
1	0.2667	0.2667	0.2667	0.3333	0.4000	0.6000
9	0.2000	0.2000	0.2000	0.2667	0.3333	0.4000
3	0.2000	0.2000	0.2000	0.2000	0.2667	
7	0.1333	0.1333	0.2000	0.2000		
2	0.0667	0.1333	0.1333			
12	0.0667	0.0667				
15	0.0667					

x_i	$P(x_i)$	Code										
1	0.2667	10	0.267	10	0.267	10	0.333	11	0.400	0	0.600	1
9	0.2000	01	0.200	01	0.200	01	0.267	10	0.333	11	0.400	0
3	0.2000	00	0.200	00	0.200	00	0.200	01	0.267	10		
7	0.1333	110	0.133	110	0.200	111	0.200	00				
2	0.0667	1110	0.133	1111	0.133	110						
12	0.0667	11111	0.067	1110								
15	0.0667	11110										

There are a total of 7 different levels of intensity, which can be represented in 3 bits. Thus, prior to compression, the total size would be $15 \times 3 = 45$bits.

The compressed image size $= 2 \times 4 + 2 \times 3 + 2 \times 3 + 3 \times 2 + 4 \times 1 + 5 \times 1 + 5 \times 1 = 40$bits.

Therefore, compression ratio $= \frac{45}{40} = 1.125$.

2) Adaboost Learning Algorithm

a)

$$\vec{y} = \begin{bmatrix}1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1\end{bmatrix};$$

$$\begin{aligned}
h_1 &= \begin{bmatrix}1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1\end{bmatrix};\\
h_2 &= \begin{bmatrix}-1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1\end{bmatrix};\\
h_3 &= \begin{bmatrix}1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1\end{bmatrix};\\
h_4 &= \begin{bmatrix}-1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1\end{bmatrix};\\
h_5 &= \begin{bmatrix}-1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1\end{bmatrix};
\end{aligned}$$

Iteration 1

$$\text{Initialize } w_i = \frac{1}{9}.$$

$$\begin{aligned}
e_{h_1} &= \frac{4}{9}\\
e_{h_2} &= \frac{6}{9}\\
e_{h_3} &= \frac{3}{9}\\
e_{h_4} &= \frac{7}{9}\\
e_{h_5} &= \frac{6}{9}
\end{aligned}$$

Best error $= \frac{3}{9}$. Thus best weak classifier $= h_3$.

$$\text{Set } \alpha_1 = \frac{1}{2}\ln\frac{1-e_{h_3}}{e_{h_3}} = 0.34657359$$

$$\begin{aligned}
w_i &= w_i e^{-\alpha_1 y_i h_k(\vec{x}_i)}\\
w_i &= \begin{cases}0.07856742, & \text{if } y_i h_k(\vec{x}_i) = 1\\ 0.15713484, & \text{otherwise}\end{cases}
\end{aligned}$$

Now we will normalize the weightings.

$$\begin{aligned}
\sum_i w_i &= 0.942809041\\
\text{Set } s = \frac{1}{\sum_i w_i} &= 1.060660172\\
w_i &= w_i s\\
\vec{w} &= \begin{bmatrix}0.08\dot{3} & 0.08\dot{3} & 0.08\dot{3} & 0.1\dot{6} & 0.1\dot{6} & 0.1\dot{6} & 0.08\dot{3} & 0.08\dot{3} & 0.08\dot{3}\end{bmatrix}
\end{aligned}$$

Iteration 2

$$
\begin{aligned}
e_{h_1} &= 0.41\dot{6} \\
e_{h_2} &= 0.75 \\
e_{h_3} &= 0.5 \\
e_{h_4} &= 0.\dot{6} \\
e_{h_5} &= 0.58\dot{3}
\end{aligned}
$$

Best error $= 0.41\dot{6}$. Thus best weak classifier $= h_1$.

$$\text{Set } \alpha_2 = \frac{1}{2} \ln \frac{1 - e_{h_1}}{e_{h_1}} = 0.16823612$$

We stop the iteration, and choose h_3 with weight α_1 and h_1 with weight α_2 for the strong classifier H.

$$
\begin{aligned}
H(\vec{x}) &= \operatorname{sgn}\left(\alpha_1 h_3(\vec{x}) + \alpha_2 h_1(\vec{x})\right) \\
&= \operatorname{sgn}\left(0.34657359 h_3(\vec{x}) + 0.16823612 h_1(\vec{x})\right)
\end{aligned}
$$

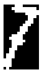
b)

$$
\begin{aligned}
H(\vec{y}) &= \operatorname{sgn}\left(0.34657359 h_3(\vec{y}) + 0.16823612 h_1(\vec{y})\right) \\
&= \operatorname{sgn}(0.34657359 \begin{bmatrix} 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 \end{bmatrix} + \\
&\quad 0.16823612 \begin{bmatrix} 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}) \\
&= \begin{bmatrix} 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 \end{bmatrix}
\end{aligned}
$$

II. Programming Tasks

1) Digital Segmentation

   a) Image 1

Image	Ground Truth	Predicted
	1	1
	2	2
	4	4
	7	1
	6	6
	7	1
	3	3
	8	8
	9	1
	5	5
	2	2
	4	8
	8	8
	1	1

Number of incorrectly identified: 4

Total: 14

Error: 0.2857

Accuracy: 0.7143

b) Image 2

Image	Ground Truth	Predicted
	1	1
	2	2
	3	3
	4	4
	1	1
	9	7
	8	8
	6	6
	2	2
	0	0
	1	1
	9	7

Number of incorrectly identified: 2

Total: 12

Error: 0.1667

Accuracy: 0.8333

c) Image 3

Image	Ground Truth	Predicted
	2	2
	0	0
	1	1
	9	1
	1	1
	0	0
	2	2
	5	5
	2	2
	0	0
	1	1
	9	1
	1	1
	1	1
	0	0
	1	1

Number of incorrectly identified: 2

Total: 16

Error: 0.1250

Accuracy: 0.8750

The segmentation method is as follows: The image is first filtered to to the state where the there are only digits left in the image and a binary image is taken from this. The digits are white and the rest is black. The image is then divided into image row segments of digits, by extracting consecutive non-zero rows. Then in each image row segment, a column scans from left to right, once there is at least a white pixel under the column, BFS will be run, to transfer the connected picels to a new blank image the size of the segment. The transferred image is then trimed to remove redundant black borders to become the required digit image.

d) Classification via Adaboost

For the accuracies, please refer the the tables above.

For the weak classifiers, I have constructed 3, using MatLab's built in functions. 2 of them are CNN networks and one is an MLP network.