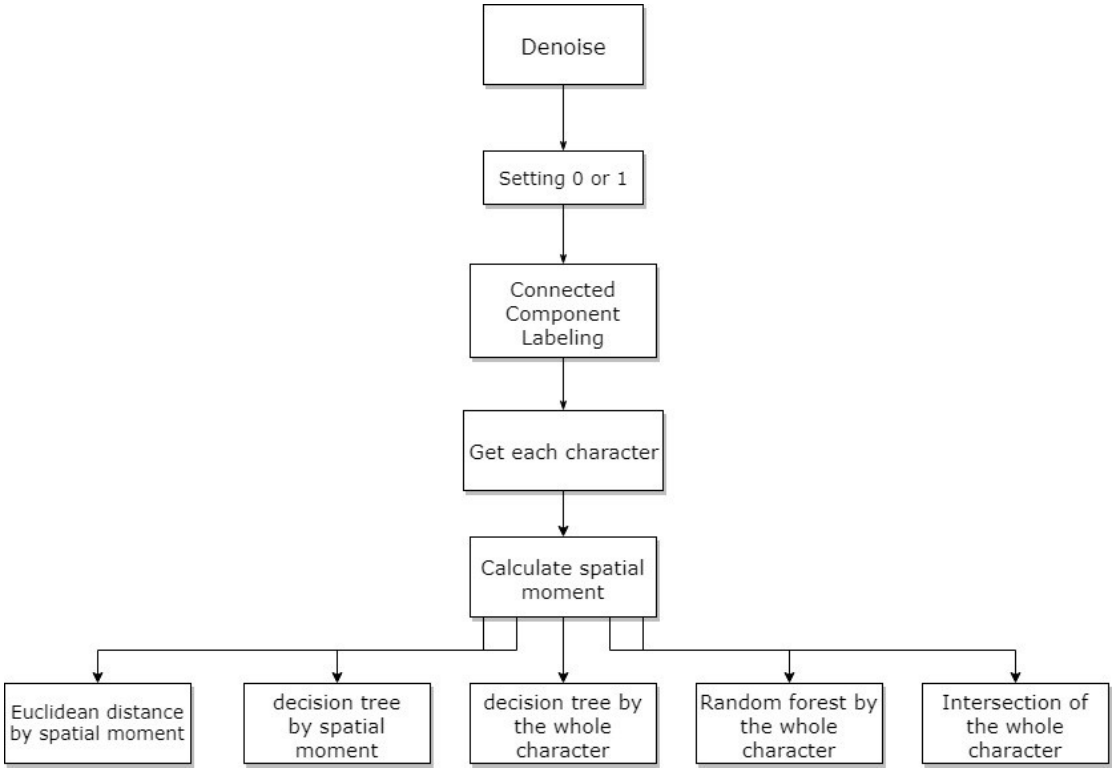
HOMEWORK ASSIGNMENT #4

Optical Character Recognition (OCR)

B04902099 資工三 黃嵩仁

PROBLEM 1: Optical Character Recognition (OCR)

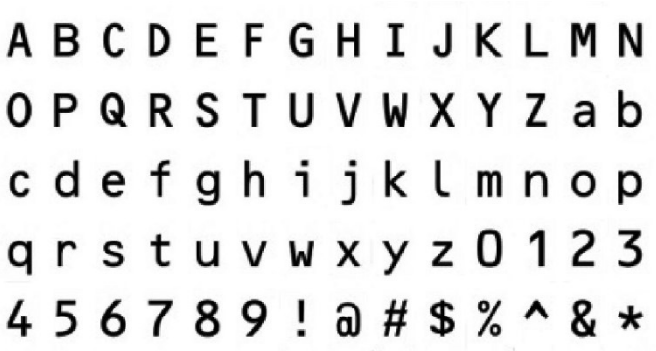
Flowchart：



Pre-processing：

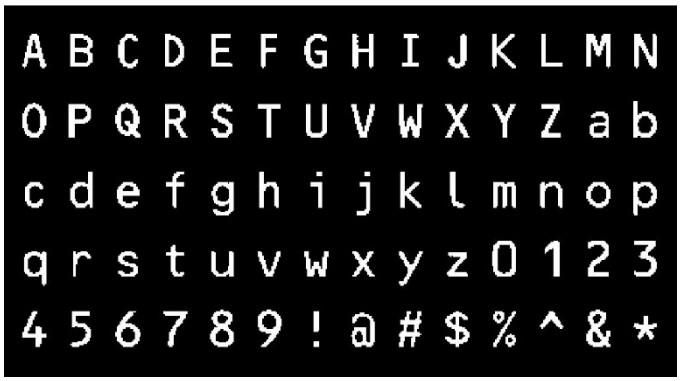
Training Part

Original image：



Step1：

設定一個threshold，來判斷是否為我們所要的object (character)。(我設定threshold為100，大於者為背景=>設為0，小於者為object=>設為1)。



Step2：

計算connected component，並對每個object (character)下label。

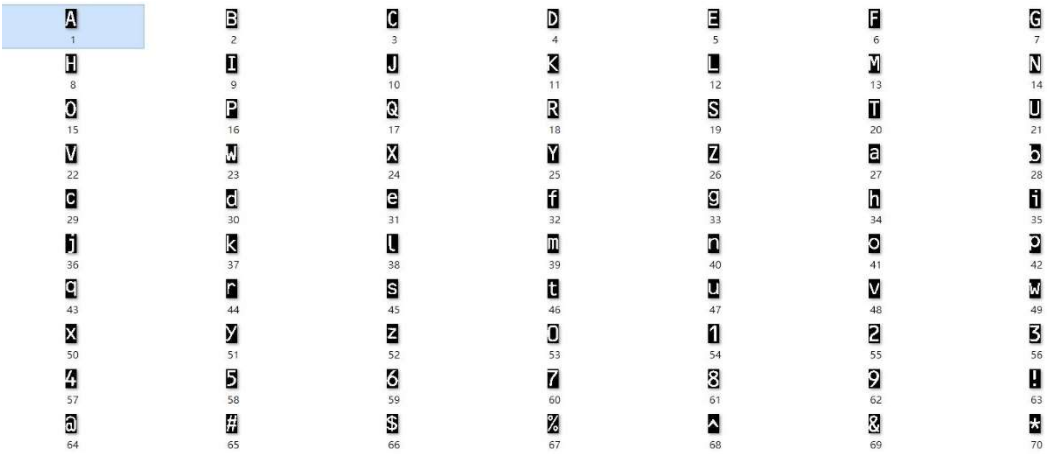
(備註1：不同於lecture4中的原版Connected Component Labeling，因為考慮像是'i', 'j', '%', '!'等character可能會被判斷成2個以上的object，因此將判定"connected"的範圍拉大，由距離為1的點擴大為距離10的點。)

(備註2：因為考慮每個character的最高位置不同，因此選擇從row = 30,75,125,175,210，五個橫切面看)

Step3：

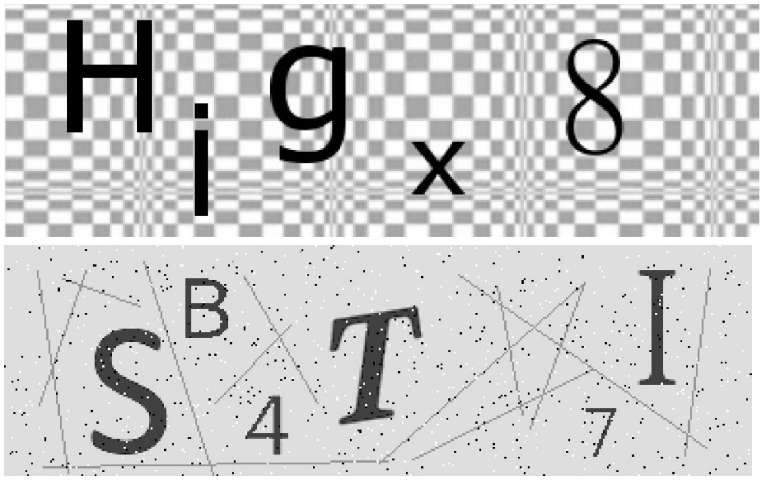
藉由labeling的結果，可判斷每個character的邊界，並切出每個character，如下圖。

(備註：我將所有character都padding到一樣的大小：height = 29, width = 17)



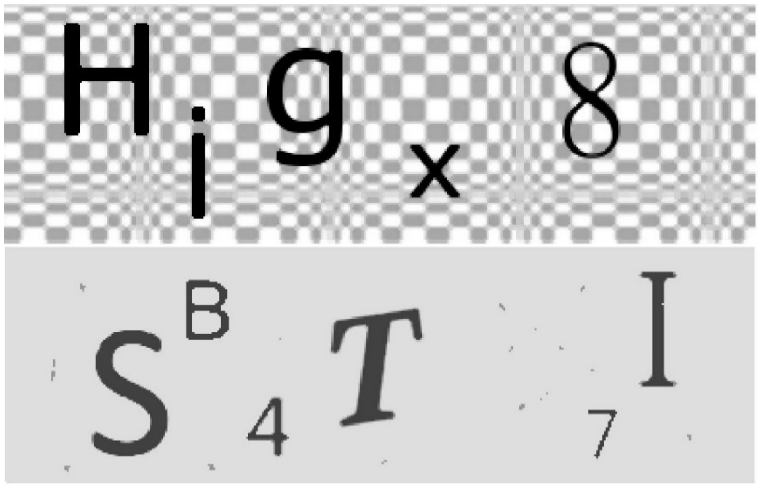
Testing Part

Original image：



Step1：

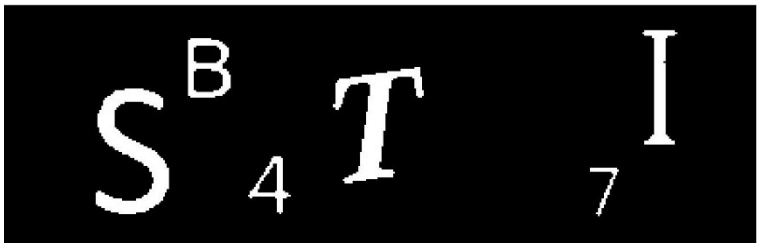
因為觀察到sample2有salt & pepper noise，因此將讀進來的image先通過median filtering，以去除noise。(採用3*3的mask，odd boundary extension)。



(備註：可觀察到sample2經過median filtering後，不只salt & pepper noise被消去，連部分灰色線條也被消除。)

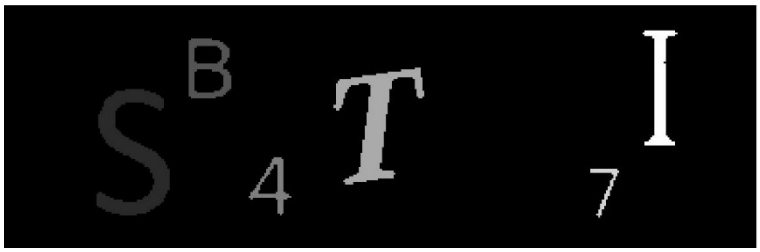
Step2：

同Training Part pre-processing 的 Step1。



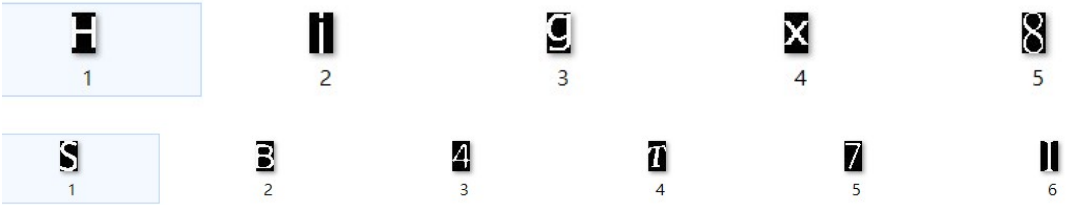
Step3：

同Training Part pre-processing 的 Step2。



Step4：

類似於Training Part pre-processing 的 Step3，但因為testing data的大小差異極大，因此在取得character的邊界後，將其縮放、padding至與training data相同的大小。(height = 29, width = 17)



Training & prediction：

Training Part

依照上課講義Lecture7 P.23中的公式，$$M(m,n)=\frac{1}{J^m K^n}\sum_{j=1}^{J}\sum_{k=1}^{K}(x_j)^m(y_k)^n F(j,k)$$

計算出M(0,0), M(1,0), M(0,1), ……, M(0,3)等10個spatial moment作為判斷的依據，如下圖。

	1	2	3	4	5	6	7	8	9	10
1	157	83.3103	78.8235	42.1684	48.0069	51.1938	27.0043	26.1374	32.4905	34.3388
2	200	100.7241	99.7059	50.5882	65.5121	61.2378	33.4366	30.6302	48.8225	41.4801
3	124	64.5172	57.9412	30.2657	32.9100	41.4566	17.2828	20.2548	21.4344	29.7102
4	158	80.5517	75.5882	38.9878	45.3391	49.0131	23.5861	23.4899	30.9699	33.0602
5	156	78.2759	61.5294	31.1420	32.4913	49.1914	16.6459	20.1685	20.1803	34.6618
6	113	48.7241	47.2941	18.7160	23.2664	26.4102	8.4039	9.4849	13.1590	16.3964
7	172	92	92.4118	50.3874	61.8651	59.0131	34.2218	33.2224	46.5575	41.6339
8	167	85.7586	79.9412	40.7748	53.5606	51.5470	27.1760	24.3639	41.1018	34.1572
9	110	56.3448	58.8235	29.9412	33.5017	36.7372	17.0508	19.4704	20.1209	26.7525
10	117	68.1379	72.2941	39.4219	48.9100	45.5719	25.5186	25.1374	34.7004	32.8570
11	155	79.4483	58.4118	30.6430	33.3806	48.5660	17.9455	19.7536	22.4973	32.9347
12	99	60.3103	31.0588	21.8884	14.7682	42.2866	11.4987	16.7797	9.1752	31.7579
13	215	96.2414	107.7647	48.4280	75.8754	52.0702	34.8769	26.3315	59.9198	32.0397
14	198	101.2759	104.4706	55.3813	73.5433	60.6314	39.6644	33.9682	58.2841	40.0335
15	169	89.5517	83.1176	43.8783	55.1038	56.1795	29.0397	27.3891	41.0615	38.9875
16	166	71	70.7647	27.2860	42.3080	37.2140	15.2784	12.6933	30.2497	22.2671
17	200	108.1034	106.5294	59.6369	71.7266	67.8561	41.0728	38.6227	53.5186	46.5167
18	179	84.7241	83.8824	40.0467	51.5294	48.7289	25.0637	23.2553	35.7444	31.4413
19	155	83.0690	79.2353	43.2799	48.4256	53.6040	26.8647	28.2210	32.8852	38.3850
20	102	39.2069	54	20.7566	31.6263	20.7574	11.5625	10.9892	19.9601	13.1874
21	147	78.1379	73.1765	38.8012	49.7370	49.1534	25.6386	24.3551	38.0643	33.9580
22	142	69.4483	73.0588	35.8864	46.2630	40.4828	21.4970	20.9230	32.7539	26.2042
23	199	107.1724	95.4118	51.1582	62.0900	65.8740	32.1018	31.4320	45.3711	43.6862
24	153	76.6552	75.4706	38.2150	45.1661	46.1891	23.1559	23.1457	30.0963	30.9118
25	109	45.8966	56.2353	23.8783	33.0588	24.0392	13.3762	12.5838	21.1512	14.6163
26	133	67.8621	68.1765	33.8296	39.6851	44.2687	19.3717	21.7399	25.2178	32.2636
27	130	69.8621	73.5294	39.1136	48.3945	41.6742	25.7259	23.1172	34.8414	26.7058
28	177	101.4483	72.6471	44.2049	45.8581	66.6159	28.4258	29.5253	33.8125	47.5477
29	99	51.1379	45.5294	23.5882	25.3979	29.9988	13.1661	14.0613	16.0509	19.1711
30	161	90.5862	97.7647	52.5923	72.7889	58.8264	38.0396	33.6517	59.1415	41.6787
31	142	72.2759	71.6471	36.2231	44.5813	40.6659	22.3749	20.3825	31.2567	24.7499
32	106	46.4828	53.8824	22.7059	29.5640	25.3746	11.8074	12.1930	17.3071	15.9573
33	176	86.2759	103.5294	50.5497	73.3633	51.7669	35.4692	30.2263	57.1928	34.7279
34	141	74.4828	66.6471	37.2880	41.8374	45.5815	24.5516	23.3732	30.5774	30.6640
35	78	39	46.5882	23.8479	28.7474	23.5113	14.9692	14.6956	18.1441	15.7730
36	104	60.0345	65.5882	36.7688	42.9931	44.0488	23.6629	26.2054	28.9593	35.8091

	1	2	3	4	5	6	7	8	9	10
37	138	76.7241	50.7059	30.2718	25.8754	49.0927	16.2950	20.2342	15.8380	34.1052
38	100	51.8966	37.6471	21.2596	15.6263	33.1213	9.6682	14.5017	7.1915	23.5409
39	163	80.4138	85.5294	42.3529	57.9862	44.4495	28.9887	23.4533	44.2316	26.7476
40	118	56.6552	60.2941	29.3408	42.9723	30.6457	21.5730	16.0503	34.7834	18.1749
41	126	64.8276	63.7647	32.9331	42.4775	37.3436	21.9380	19.0404	31.7790	23.2999
42	168	75	71.4118	30.3063	45.7855	40.3817	19.0613	15.2601	33.9792	24.2520
43	160	73.6552	94.8824	45.4341	69.9619	40.7301	34.3330	26.2303	56.4631	25.1449
44	70	29.4483	25.8235	9.7039	12.4048	14.1237	4.2737	4.0992	7.1958	7.6545
45	114	59.6897	60.9412	32.5030	37.9446	35.0166	20.5753	19.3263	25.9792	22.2857
46	95	50.3103	45.9412	25.2150	24.2803	30.9501	13.7467	16.0574	13.9371	21.1242
47	121	67.5517	66.5882	37.6349	49.2042	41.2592	27.4499	23.2128	40.4059	26.8231
48	100	51.5862	52.6471	27.2089	33.9965	29.2937	16.8872	15.4735	24.5705	17.8977
49	142	77.9655	71.1765	39.0162	46.9204	46.3508	24.9773	23.1627	34.7661	29.1593
50	113	58.4483	58.3529	30.4503	35.6471	33.6064	18.7150	17.6433	24.0725	20.8804
51	114	53.2069	50	21.1988	28.4360	30.4388	10.6664	10.8961	18.7165	19.9061
52	114	58.2759	57.0588	28.5091	33.0242	33.9738	16.3342	16.3736	21.0222	21.5641
53	191	96	96	48.3489	67.0173	60.9727	33.8967	30.9787	52.9363	43.4288
54	120	54.1034	73.8235	34.6673	47.7336	31.0999	23.0091	20.4998	31.8370	20.6610
55	162	82.4138	80.9412	39.2252	49.6955	54.6136	23.3500	25.0441	34.0818	40.6534
56	168	80.5517	98.2941	47.6552	65.7682	51.3698	32.3568	29.8514	47.4107	37.2513
57	142	77.5862	69.1176	39.4442	40.8893	47.8121	24.4069	25.2680	26.8245	31.4451
58	147	69.6207	75.4706	37.1278	46.2180	43.1712	23.3368	23.0515	31.3346	30.4065
59	177	96.4483	90.6471	49.5680	58.4048	61.3234	32.6726	31.8740	42.5176	42.5944
60	123	47.3103	61.1765	20.9959	35.9308	26.0927	10.6960	10.3730	23.5766	17.1942
61	209	111.1724	107.2941	56.9838	69.1696	71.8906	37.2281	36.9366	49.9393	51.7601
62	175	82.0345	87.1765	40.6714	55.3356	47.1998	25.0909	22.9310	39.2400	30.7699
63	71	31.3103	35.4118	15.6065	17.9654	18.2307	7.9124	9.0885	9.2636	12.5721
64	182	101.8276	97.3529	54.7241	65.8720	66.7503	37.1417	35.7602	49.8817	47.6726
65	178	87.8621	94.3529	43.6207	60.2699	51.6576	26.6893	24.4183	42.8345	33.7189
66	184	91.5517	91.5882	46.9878	55.8374	55.0737	29.2847	28.7239	37.9448	36.7992
67	141	69.7241	73.4118	37.7120	46.9273	44.1902	24.7426	24.3548	33.5490	31.4069
68	81	41.4828	39.7059	20.4544	23.6990	22.0654	12.5901	10.9503	15.7523	12.1283
69	218	123.2759	103.1765	60.2333	61.8408	83.3068	37.5152	41.7576	42.0493	61.6367
70	112	58.1379	58.5294	30.4442	35.4844	31.9715	18.5108	16.7951	23.6151	18.4428

Testing Part
True Answer：
Sample1：Higx8
Sample2：SB4T7I

法一：
若單純使用spatial moment，計算Euclidean distance來決定答案，得到：
Sample1：qiNoh
Sample2：QGvz^t
正確率：9.09%
若先將character skeletonizing，單純使用spatial moment，計算Euclidean distance來決定答案，得到：
Sample1：Mc#u&
Sample1：Q&G5YI
正確率：9.09%

法二：
若單純使用spatial moment，以及decision tree來判斷，得到：
Sample1：HiNoV
Sample2：NRtCiI
正確率：18.18%
若先將character skeletonizing，單純使用spatial moment，以及decision tree來判斷，得到：
Sample1：Bc#I#
Sample2：H#XICI
正確率：9.09%

法三：
若使用整個character (29*17)的資訊，利用decision tree來判斷，得到：
Sample1：K40mj
Sample2：jBJ*34
正確率：9.09%
若先將character skeletonizing，若使用整個character (29*17)的資訊，利用decision tree來判斷，得到：
Sample1：*ji*j
Sample2：VF**hj
正確率：0%

法四：
若使用整個character (29*17)的資訊，利用random forest來判斷，得到：
Sample1：KI!xj
Sample2：j84V!I
(因為random forest的隨機性，以及tree數量的調整，每次的結果都有可能不同，但相較於decision tree穩定且正確率較高一點，平均而言準確率在18.18%~27.17%左右，最高到過36.36%)

法五：
在嘗試過以上幾種方法後，想到既然我都利用pre-processing將character處理到相近的狀態，為何不直接進行比對，在計算training set character與testing character的交集後，發現做出來的結果比上面幾種方法都還要來的好。
Sample1：MI0x8
Sample2：8BQ$7I
正確率：45.45%
(最後選擇法五)

討論：

由上述五種方法，可發現這個範例下，使用spatial moment來分類並不是個好選項，可能是因為character這種object過於纖細，細節過多，單用10個spatial moment無法表現出其真正的差異。相反的，因為pre-processing的處理還算不錯，因此直接將testing data與training dataset做交集的這種基礎做法，分類的程度還比spatial moment來的好。此外可發覺，法五中錯誤的地方'l'與'i'、 'S'與'8'等組合輪廓都極為相近，但若是直接考慮connected component數與hole的數量，又會遇到像是training data與testing data中 '4' 的問題(training中hole = 0，testing中hole = 1)，仍需再思考更好的方法來實作。