Python extract_counts protocol

Daniëlla Zinn, daniellazinn@hotmail.nl

This script gathers the counted cells per ImageJ Point Tool counter for every brain slice and writes them into one CSV file. This script with the output from the merge_csv.py script, but also works on an Excel file with multiple tabs containing counting cell counts for multiple brain slices.

1. This script (extract_counts.py) works for an Excel file that has multiple sheets with original names, a column indicating which Point Tool counter was used, and a column indicating which indicates the number of cells counted (should be the same per counter), for example:

	A	B	C	D	E	F	G	H	I
1			Mean	X	Y	Counter	Count	Contrast	
2	**0**	1	142	287.167	1762.5	1	33	89	
3	**1**	2	196	300.833	1776.833	1	33	35	
4	**2**	3	197	338.833	1743.167	1	33	34	
5	**3**	4	202	352.5	1764.167	1	33	29	
6	**4**	5	204	368.5	1765.167	1	33	27	
7	**5**	6	167	422.5	1715.167	1	33	64	
8	**6**	7	99	517.5	1691.5	1	33	132	
9	**7**	8	160	581	1620	1	33	71	
10	**8**	9	195	610.5	1609	1	33	36	
11	**9**	10	210	707	1536.5	1	33	21	
12	**10**	11	199	741.5	1535.5	1	33	32	
13	**11**	12	191	763.5	1523	1	33	40	

138_10x_hyp_amy015 | 138_10x_hyp_amy016 | 138_10x_hyp_amy017 | ...

   a. Only the “Counter” and “Count” columns and name of the sheet are relevant.

2. **Download** extract_counts.py.
3. **Download** something to visualise code, such as **Visual Studio Code**, which will be used in this protocol.
4. **Personalize** the script.
   a. First, make a CSV file for saving the data.
   b. Copy the path to the CSV file and paste it after “counting_results” (replacing the text that is already there).
   c. Copy the path to the Excel sheet with all data and paste it after “path”.

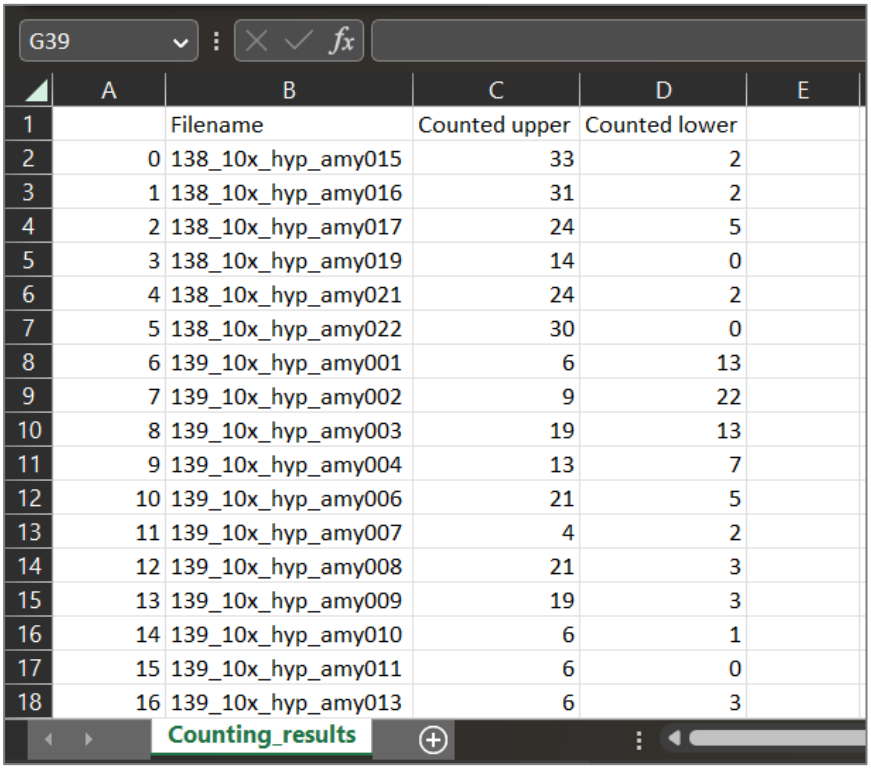
```
10    path = r"C:\Users\Danie\Documents\Internship\Counting\Merge_data.xlsx"
11    counting_results = r"C:\Users\Danie\Documents\Internship\Counting\Counting_results.csv"
```

   Since the backward slash “\” is a special character in Python strings, an “r” should be placed before the string. If the path includes forward slashes “/”, the “r” can be removed.
   d. The script is based on a script for counting the upper and lower blade of the dentate gyrus. Cells in the upper blade were counted with Counter 1 from the ImageJ Point Tool, and the lower blade with Counter 2. You can choose to rename the variables “counted_upper” and “counted_lower” with what applies to your situation (lines 20, 21, 26, 31, 35, 38)
   e. If different counters were used, the numbers can be adjusted on lines 25, 26, 30 and 31.
   f. If more than two counters were used, lines 23 up until 26 can be copied, adjusted, and pasted after line 31.

5. **Run** the script (Ctrl+S) after saving the changes.
    a. In Visual Studio Code this can be done by clicking the play button in the right corner: 
    b. This can also be ran in the Visual Studio Code terminal, or any other terminal by opening the file location of the script and typing “python extract_counts.py” and pressing enter.

```
PS C:\Users\Danie\Documents\Internship\Cellcount_and_Analysis_Fiji-ImageJ> python extract_counts.py
```

6. **Check** the CSV, it should look something like this:



	A	B	C	D	E
1		Filename	Counted upper	Counted lower	
2	0	138_10x_hyp_amy015	33	2	
3	1	138_10x_hyp_amy016	31	2	
4	2	138_10x_hyp_amy017	24	5	
5	3	138_10x_hyp_amy019	14	0	
6	4	138_10x_hyp_amy021	24	2	
7	5	138_10x_hyp_amy022	30	0	
8	6	139_10x_hyp_amy001	6	13	
9	7	139_10x_hyp_amy002	9	22	
10	8	139_10x_hyp_amy003	19	13	
11	9	139_10x_hyp_amy004	13	7	
12	10	139_10x_hyp_amy006	21	5	
13	11	139_10x_hyp_amy007	4	2	
14	12	139_10x_hyp_amy008	21	3	
15	13	139_10x_hyp_amy009	19	3	
16	14	139_10x_hyp_amy010	6	1	
17	15	139_10x_hyp_amy011	6	0	
18	16	139_10x_hyp_amy013	6	3	

Counting_results