

Лабораторна робота №3
"Реалізація алгоритму оптимізації роєм часток
для пошуку глобального мінімуму функції."

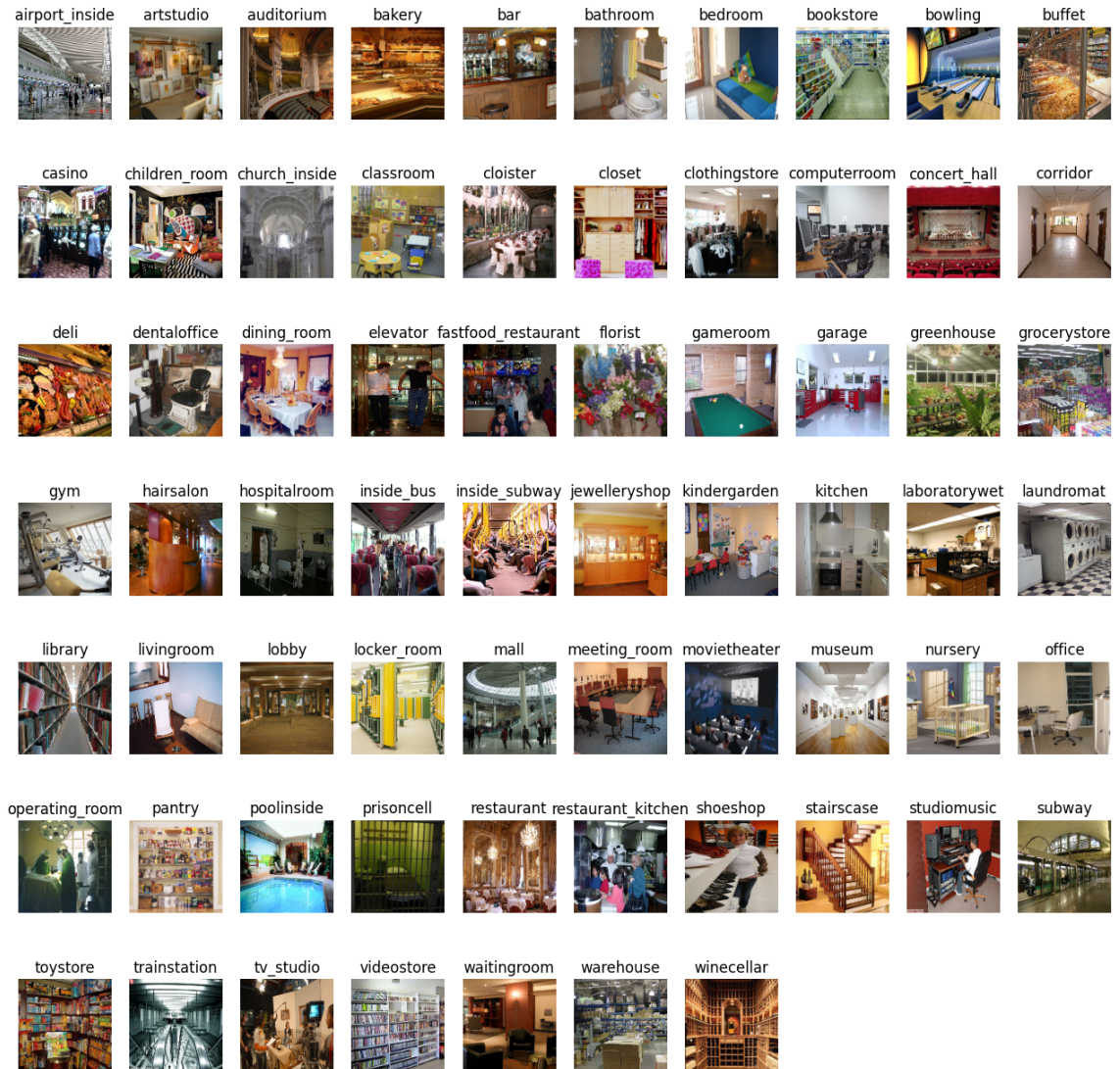
Роботу виконав:
Климентьев Максим
3-го курсу
групи ФІ-21

Зміст

1	Опис набору даних. Візуалізація прикладів з набору даних.	3
2	Опис базової архітектури.	4
2.1	Архітектура нейронної мережі, кількість параметрів.	4
2.2	Точність класифікації.	9
2.3	Візуалізація функцій втрат (тестовий та тренінговий набір).	13
3	Опис експериментів.	17
3.1	Які зміни було внесено.	17
3.2	Висновки: як це вплинуло на результат, процес тренування,	17
4	Результати порівняння.	17
5	Висновки.	17

1 Опис набору даних. Візуалізація прикладів з набору даних.

67 внутрішніх помешкань — від Аеропорту до Винного погребу



2 Опис базової архітектури.

2.1 Архітектура нейронної мережі, кількість параметрів.

1. 1 Conv2D 32:

Layer (type)	Output Shape	Param
conv2d	(None, 128, 128, 32)	896
batch_normalization	(None, 128, 128, 32)	128
max_pooling2d	(None, 32, 32, 32)	0
flatten	(None, 32768)	0
dense	(None, 256)	8,388,864
dropout	(None, 256)	0
dense	(None, 67)	17,219

Total params: 8,407,107 (32.07 MB)

Trainable params: 8,407,043 (32.07 MB)

Non-trainable params: 64 (256.00 B)

2. 2 Conv2D 32-32:

Layer (type)	Output Shape	Param
conv2d	(None, 128, 128, 32)	896
batch_normalization	(None, 128, 128, 32)	128
max_pooling2d	(None, 32, 32, 32)	0
conv2d	(None, 32, 32, 32)	9,248
batch_normalization	(None, 32, 32, 32)	128
max_pooling2d	(None, 8, 8, 32)	0
flatten	(None, 2048)	0
dense	(None, 256)	524,544
dropout	(None, 256)	0
dense	(None, 67)	17,219

Total params: 552,163 (2.11 MB)

Trainable params: 552,035 (2.11 MB)

Non-trainable params: 128 (512.00 B)

3. 3 Conv2D 32-32-32:

Layer (type)	Output Shape	Param
conv2d	(None, 128, 128, 32)	896
batch_normalization	(None, 128, 128, 32)	128
max_pooling2d	(None, 32, 32, 32)	0
conv2d	(None, 32, 32, 32)	9,248
batch_normalization	(None, 32, 32, 32)	128
max_pooling2d	(None, 8, 8, 32)	0
conv2d	(None, 8, 8, 32)	9,248
batch_normalization	(None, 8, 8, 32)	128
max_pooling2d	(None, 2, 2, 32)	0
flatten	(None, 128)	0
dense	(None, 256)	33,024
dropout	(None, 256)	0
dense	(None, 67)	17,219

Total params: 70,147 (274.40 KB)

Trainable params: 70,019 (273.91 KB)

Non-trainable params: 128 (512.00 B)

4. 3 Conv2D 32-64-128:

Layer (type)	Output Shape	Param
conv2d	(None, 128, 128, 32)	896
batch_normalization	(None, 128, 128, 32)	128
max_pooling2d	(None, 32, 32, 32)	0
conv2d	(None, 32, 32, 64)	18,496
batch_normalization	(None, 32, 32, 64)	256
max_pooling2d	(None, 8, 8, 64)	0
conv2d	(None, 8, 8, 128)	73,856
batch_normalization	(None, 8, 8, 128)	512
max_pooling2d	(None, 2, 2, 128)	0
flatten	(None, 512)	0
dense	(None, 256)	131,328
dropout	(None, 256)	0
dense	(None, 67)	17,219

Total params: 242,691 (948.01 KB)

Trainable params: 242,243 (946.26 KB)

Non-trainable params: 448 (1.75 KB)

5. 6 Conv2D 32-64-128-256:

Layer (type)	Output Shape	Param #
conv2d	(None, 128, 128, 32)	896
batch_normalization	(None, 128, 128, 32)	128
max_pooling2d	(None, 64, 64, 32)	0
conv2d	(None, 64, 64, 64)	18,496
batch_normalization	(None, 64, 64, 64)	256
conv2d	(None, 64, 64, 64)	36,928
batch_normalization	(None, 64, 64, 64)	256
max_pooling2d	(None, 32, 32, 64)	0
dropout	(None, 32, 32, 64)	0
conv2d	(None, 32, 32, 128)	73,856
batch_normalization	(None, 32, 32, 128)	512
conv2d	(None, 32, 32, 128)	147,584
batch_normalization	(None, 32, 32, 128)	512
max_pooling2d	(None, 8, 8, 128)	0
dropout	(None, 8, 8, 128)	0
conv2d	(None, 8, 8, 256)	295,168
batch_normalization	(None, 8, 8, 256)	1,024
max_pooling2d	(None, 2, 2, 256)	0
flatten	(None, 1024)	0
dense	(None, 256)	262,400
dropout	(None, 256)	0
dense	(None, 67)	17,219

Total params: 855,235 (3.26 MB)

Trainable params: 853,891 (3.26 MB)

Non-trainable params: 1,344 (5.25 KB)

6. 9 Conv2D 32-64-128-256:

Layer (type)	Output Shape	Param #
conv2d	(None, 128, 128, 32)	896
batch_normalization	(None, 128, 128, 32)	128
conv2d	(None, 128, 128, 32)	9,248
batch_normalization	(None, 128, 128, 32)	128
max_pooling2d	(None, 64, 64, 32)	0
conv2d	(None, 64, 64, 64)	18,496
batch_normalization	(None, 64, 64, 64)	256
conv2d	(None, 64, 64, 64)	36,928
batch_normalization	(None, 64, 64, 64)	256
conv2d	(None, 64, 64, 64)	36,928
batch_normalization	(None, 64, 64, 64)	256
max_pooling2d	(None, 32, 32, 64)	0
dropout	(None, 32, 32, 64)	0
conv2d	(None, 32, 32, 128)	73,856
batch_normalization	(None, 32, 32, 128)	512
conv2d	(None, 32, 32, 128)	147,584
batch_normalization	(None, 32, 32, 128)	512
conv2d	(None, 32, 32, 128)	147,584
batch_normalization	(None, 32, 32, 128)	512
max_pooling2d	(None, 8, 8, 128)	0
dropout	(None, 8, 8, 128)	0
conv2d	(None, 8, 8, 256)	295,168
batch_normalization	(None, 8, 8, 256)	1,024
conv2d	(None, 8, 8, 256)	590,080
batch_normalization	(None, 8, 8, 256)	1,024
max_pooling2d	(None, 2, 2, 256)	0
flatten	(None, 1024)	0
dense	(None, 256)	262,400
dropout	(None, 256)	0
dense	(None, 67)	17,219

Total params: 1,640,995 (6.26 MB)

Trainable params: 1,638,691 (6.25 MB)

Non-trainable params: 2,304 (9.00 KB)

7. 3 Conv2D 128-256-512:

Layer (type)	Output Shape	Param #
conv2d	(None, 128, 128, 128)	3,584
batch_normalization	(None, 128, 128, 128)	512
max_pooling2d	(None, 32, 32, 128)	0
conv2d	(None, 32, 32, 256)	295,168
batch_normalization	(None, 32, 32, 256)	1,024
max_pooling2d	(None, 8, 8, 256)	0
dropout	(None, 8, 8, 256)	0
conv2d	(None, 8, 8, 512)	1,180,160
batch_normalization	(None, 8, 8, 512)	2,048
max_pooling2d	(None, 2, 2, 512)	0
dropout	(None, 2, 2, 512)	0
flatten	(None, 2048)	0
dense	(None, 1024)	2,098,176
dropout	(None, 1024)	0
dense	(None, 67)	68,675

Total params: 3,649,347 (13.92 MB)

Trainable params: 3,647,555 (13.91 MB)

Non-trainable params: 1,792 (7.00 KB)

2.2 Точність класифікації.

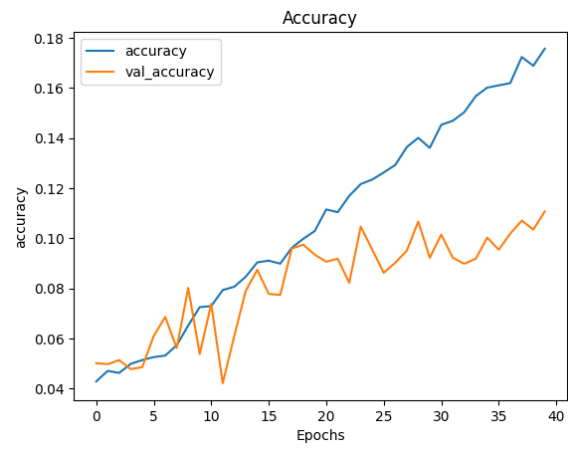


Рис. 1: 1

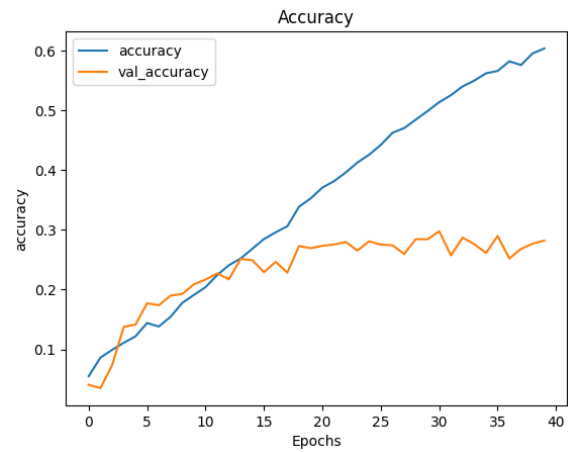


Рис. 2: 2

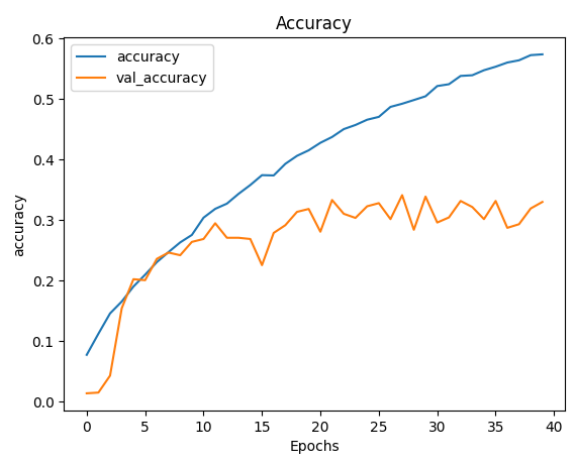


Рис. 3: 3

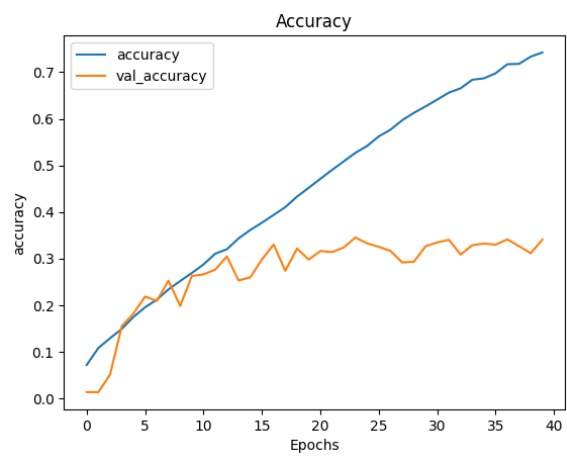


Рис. 4: 4

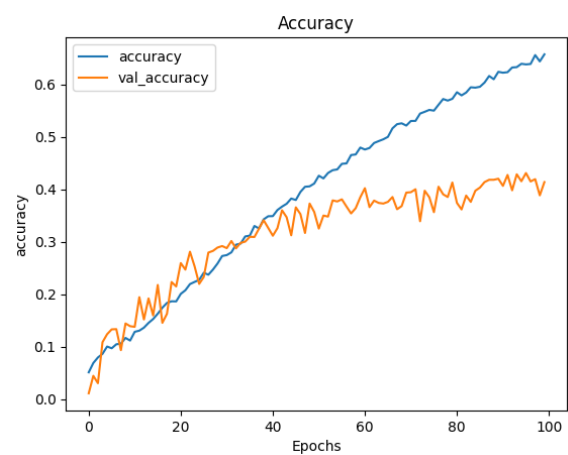


Рис. 5: 5

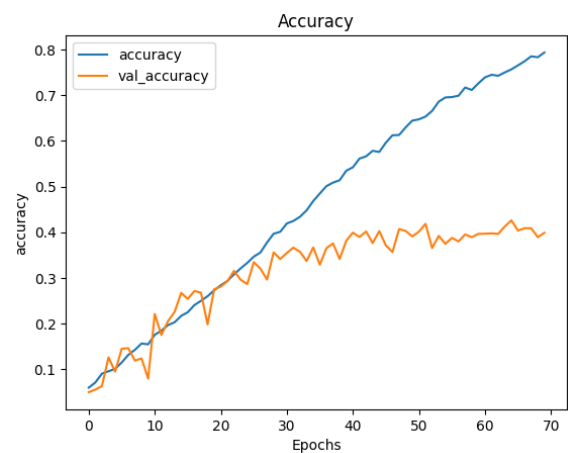


Рис. 6: 6

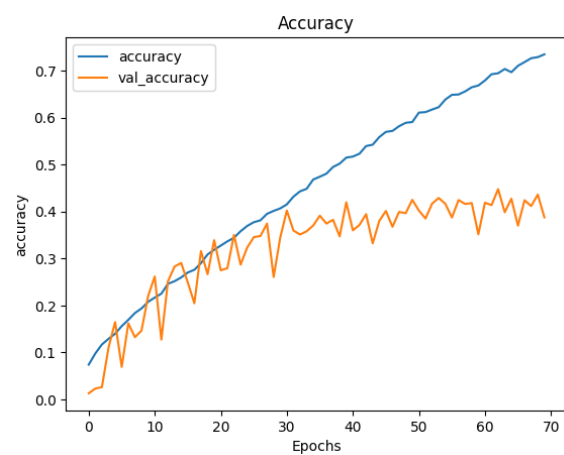


Рис. 7: 7

2.3 Візуалізація функцій втрат (тестовий та тренінговий набір).

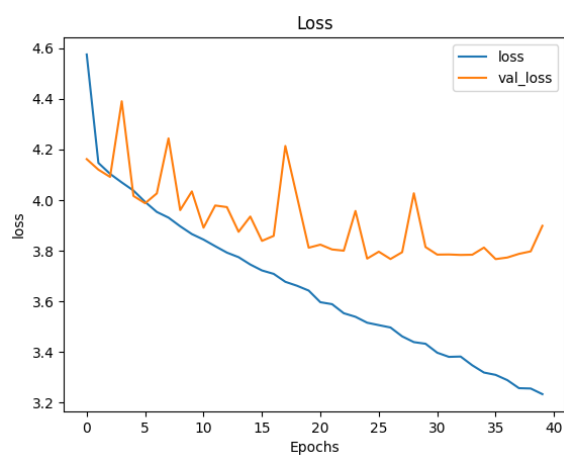


Рис. 8: 1

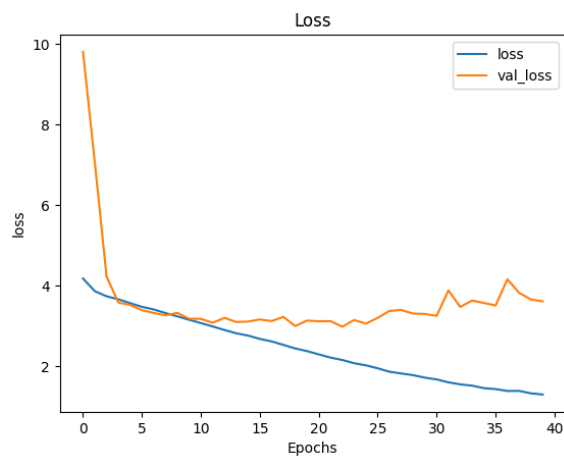


Рис. 9: 2

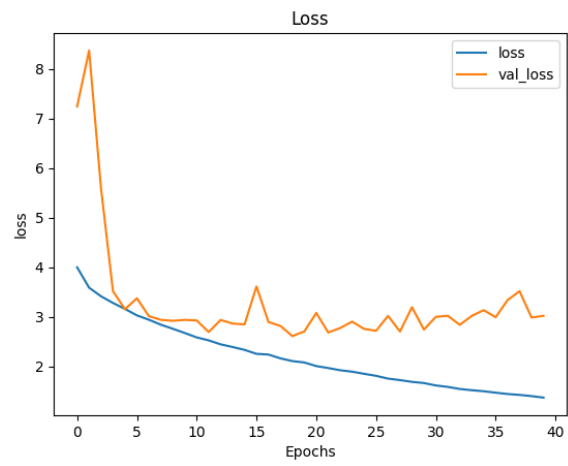


Рис. 10: 3

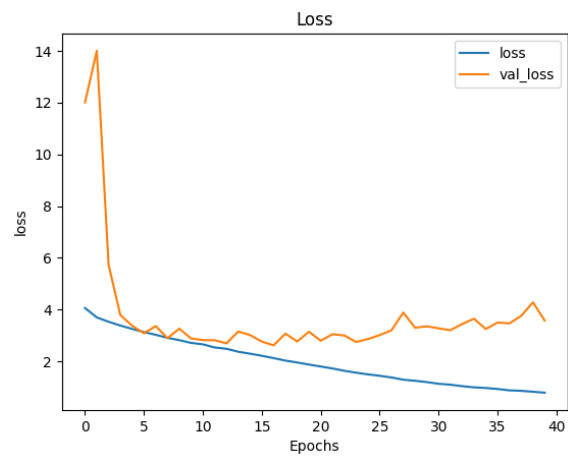


Рис. 11: 4

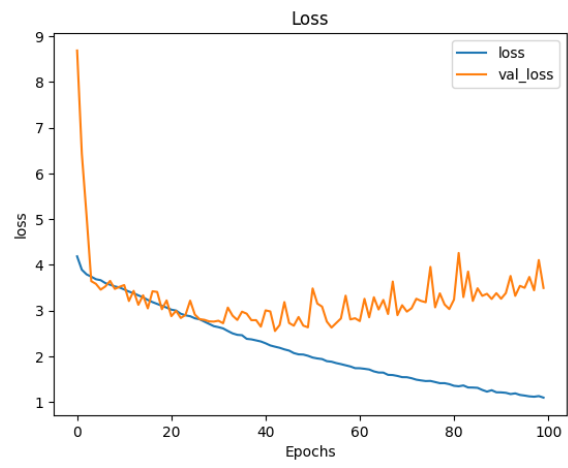


Рис. 12: 5

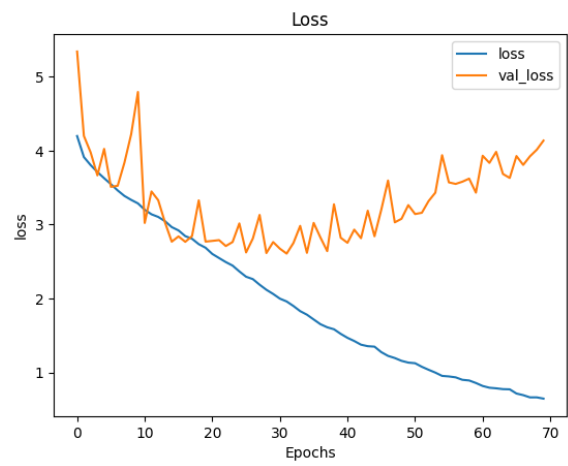


Рис. 13: 6

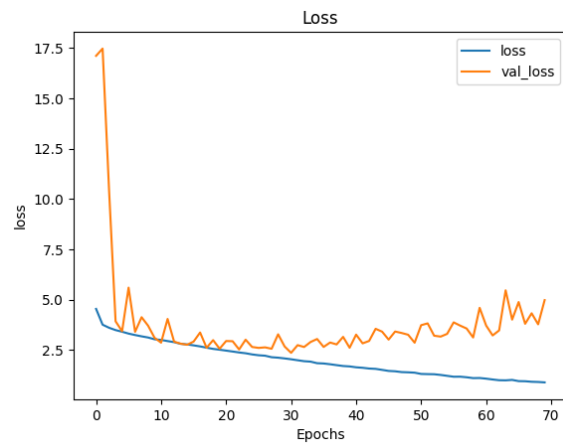


Рис. 14: 7

3 Опис експериментів.

3.1 Які зміни було внесено.

- Базова версія - 1 Conv2D 32
- Run №2 - 2 Conv2D 32-32
- Run №3 - 3 Conv2D 32-32-32
- Run №4 - 3 Conv2D 32-64-128
- Run №5 - 6 Conv2D 32-64-128-256
- Run №6 - 9 Conv2D 32-64-128-256
- Run №7 - 3 Conv2D 128-256-512

3.2 Висновки: як це вплинуло на результат, процес тренування, ...

Більше Conv2D дає більше точності, але, скоріше за все, вище 50% не отримати, треба ще використовувати щось окрім Conv2D, наприклад, збільшити розмір зображення.

4 Результати порівняння.

Опис	Conv2D	Top2	Acc	Loss	Epoch	Tr. Time
Базова	1	18	10	396	40	87
Run №2	2	38	25	363	40	77
Run №3	3	46	31	299	40	74
Run №4	3	47	33	363	40	80
Run №5	6	54	41	336	100	275
Run №6	9	52	39	419	70	468
Run №7	3	51	38	476	70	475
Best №7	3	58	45	347	63	475

5 Висновки.

Найкраще, що вдалося отримати простою моделлю - 45% точності, але з досить великими втратами.