Workout Classification using LeNet and AlexNet

Alessia Marzotti

Anuja Saira Abraham

Our original dataset

Out dataset containes images from 22 different classes of workout esercises:

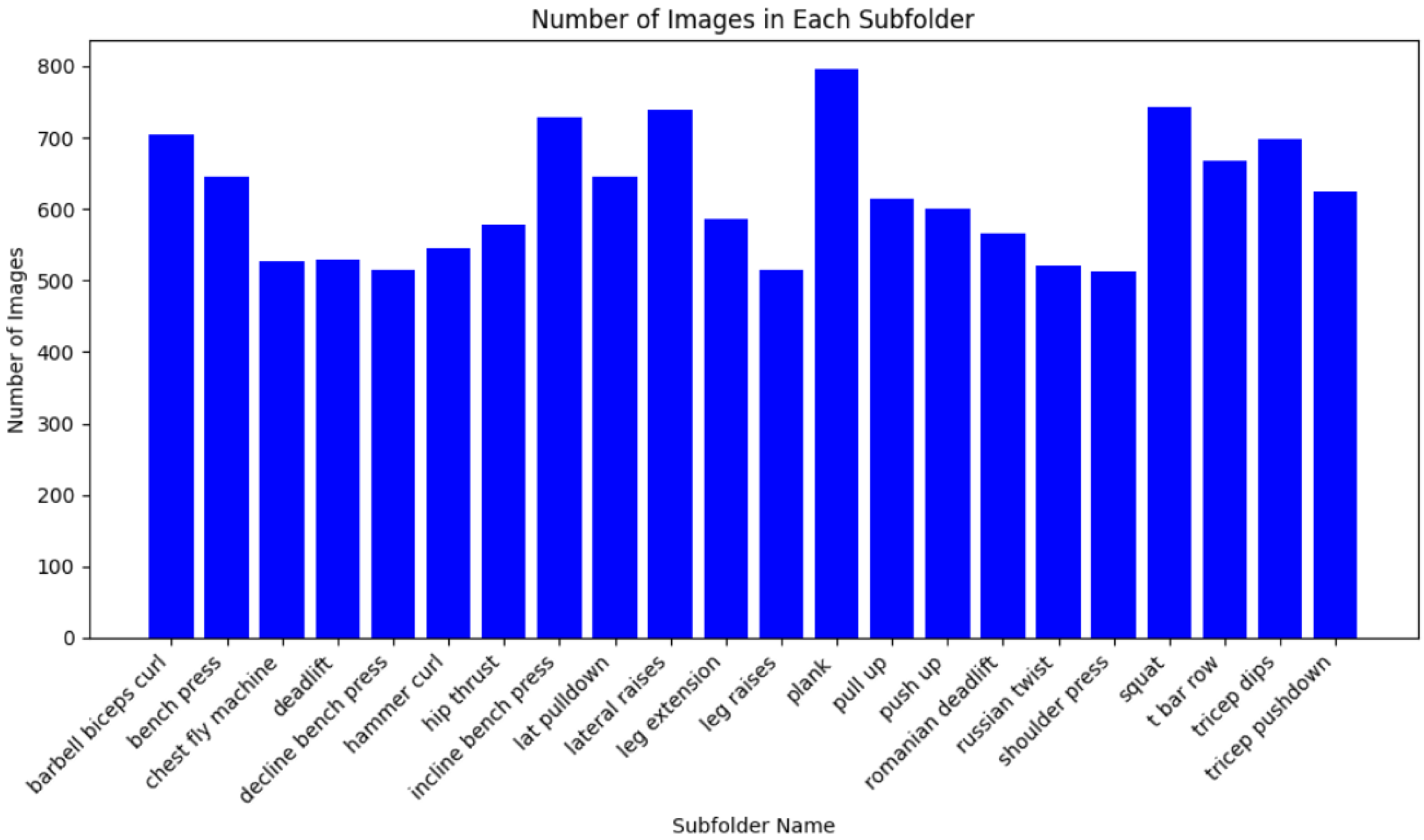
Examples: bench press, pull up, push up, leg raises, hip trust,…

Project Goal: build a model which is able to predict the workout exercise class given an image

Bench press

Pull up

Our final dataset:



Data Preparation

1

Data balancing:

We have rebalanced the dataset because some classes were having too many images compared to other classes.

2

Data splitting:

We have randomly split the data into train, validation and test (80%-10%-10%).

3

Data augmentation:

Methods implemented:

- Resize
- RandomHorizontalFlip
- Grayscale

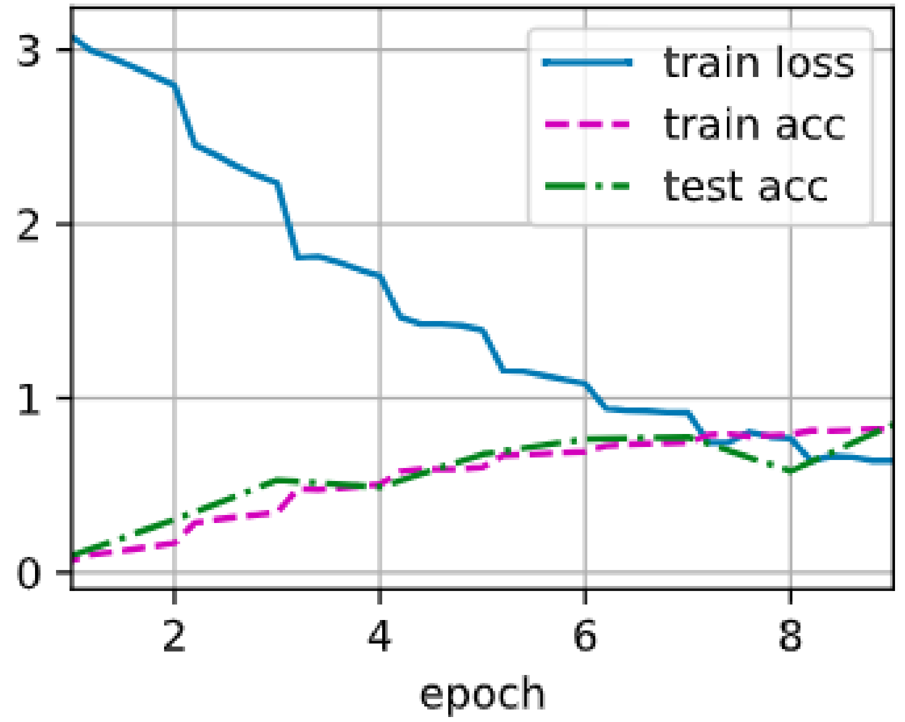
First Architecture LeNet

Model result:

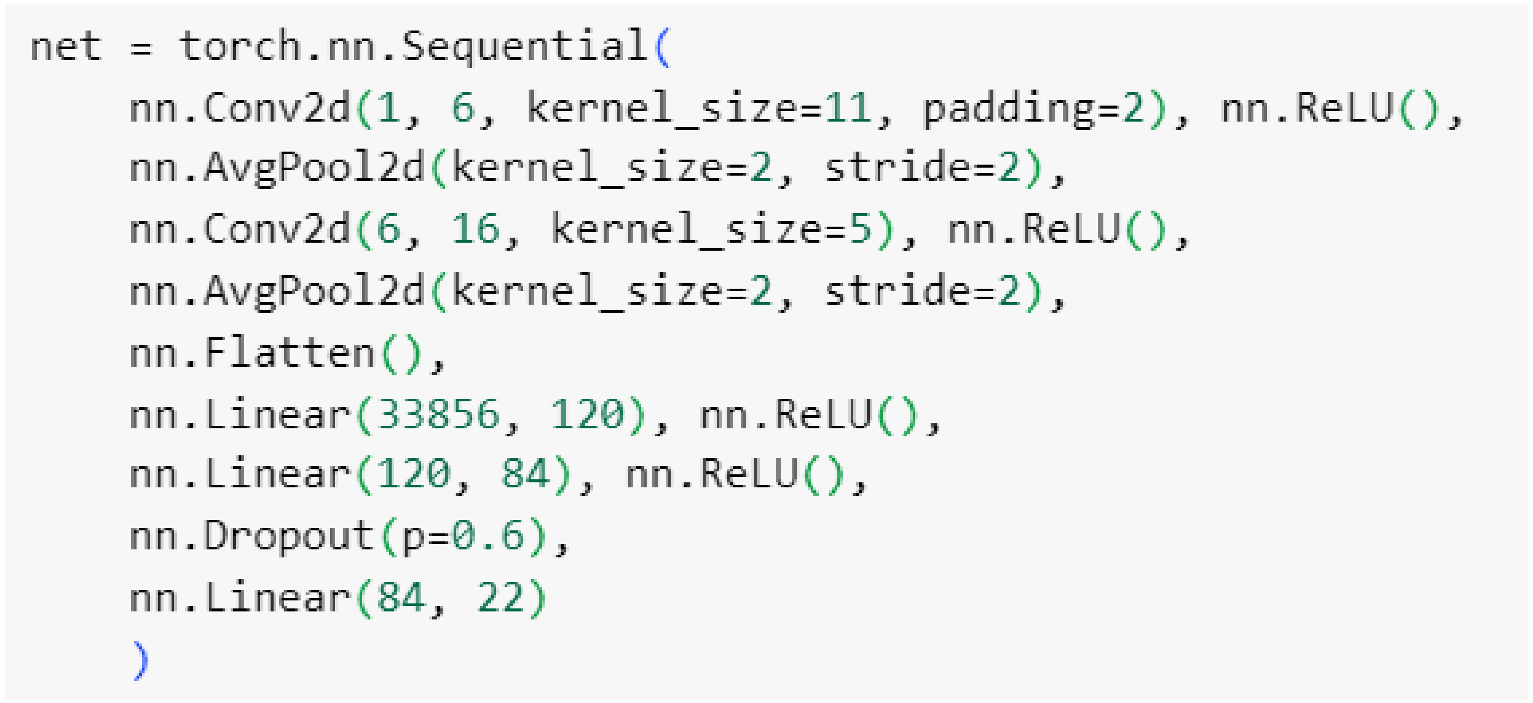
Our **implementation** is composed by:

- A 11x11 convolutional layer, followed by an average pooling
- A 5X5 convolutional layer, followed by an average pooling
- A flatten layer
- A fully connected layer 1
- A fully connected layer 2
- A dropout layer
- A fully connected layer 3

```
loss 0.642, train acc 0.823, val acc 0.851
1071.7 examples/sec on cuda:0
```



Implementation:



```
net = torch.nn.Sequential(
    nn.Conv2d(1, 6, kernel_size=11, padding=2), nn.ReLU(),
    nn.AvgPool2d(kernel_size=2, stride=2),
    nn.Conv2d(6, 16, kernel_size=5), nn.ReLU(),
    nn.AvgPool2d(kernel_size=2, stride=2),
    nn.Flatten(),
    nn.Linear(33856, 120), nn.ReLU(),
    nn.Linear(120, 84), nn.ReLU(),
    nn.Dropout(p=0.6),
    nn.Linear(84, 22)
    )
```

Notes:
- We applied a bigger kernel size
- We used a ReLU activation function
- We inserted a Dropout layer

LeNet Predictions

Some examples:

True: tricep dips
Predicted: tricep dips

True: plank
Predicted: plank

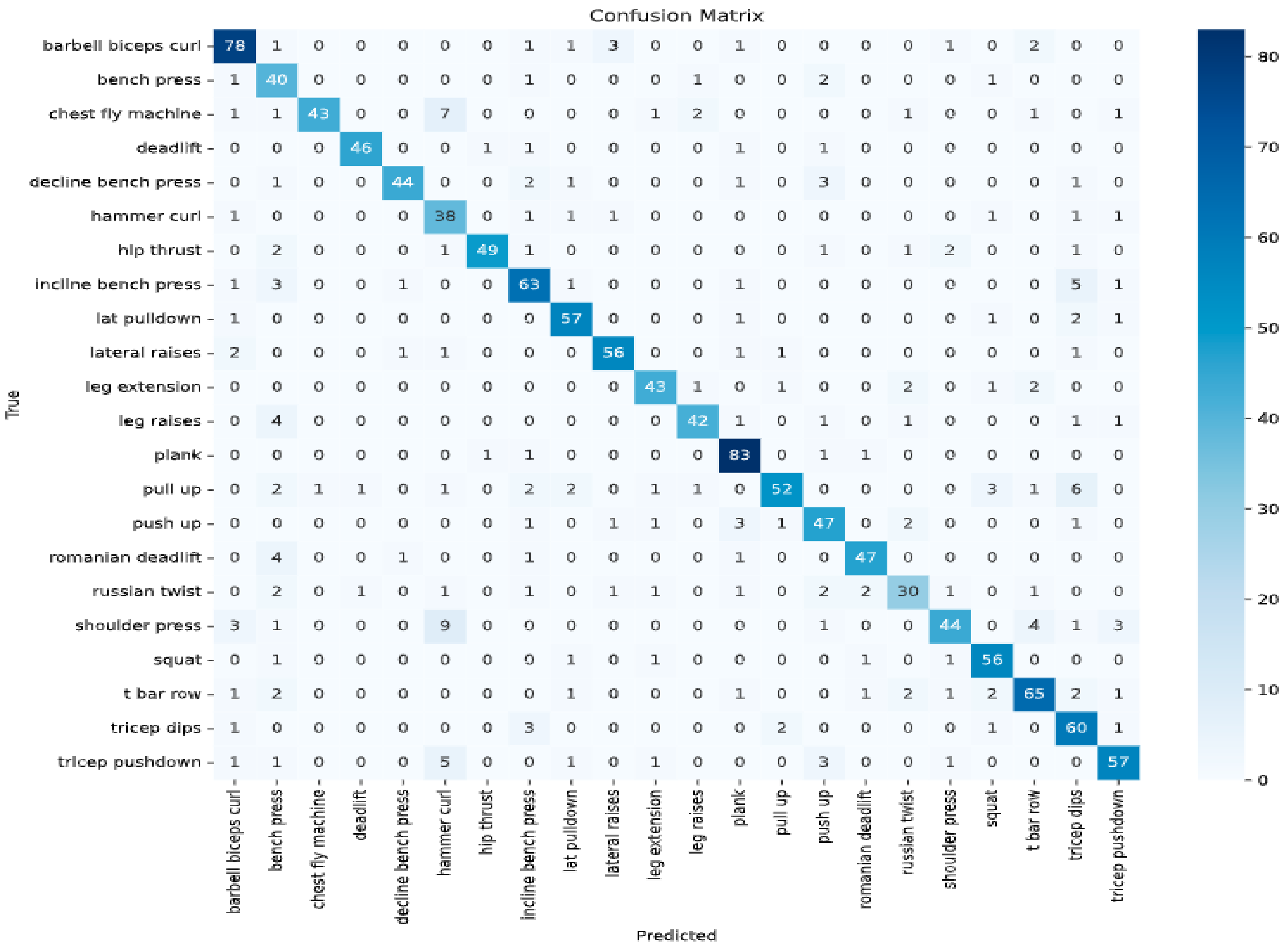
True: squat
Predicted: squat

We also created a function to *display missclassified* images:

True: tricep dips
Predicted: squat

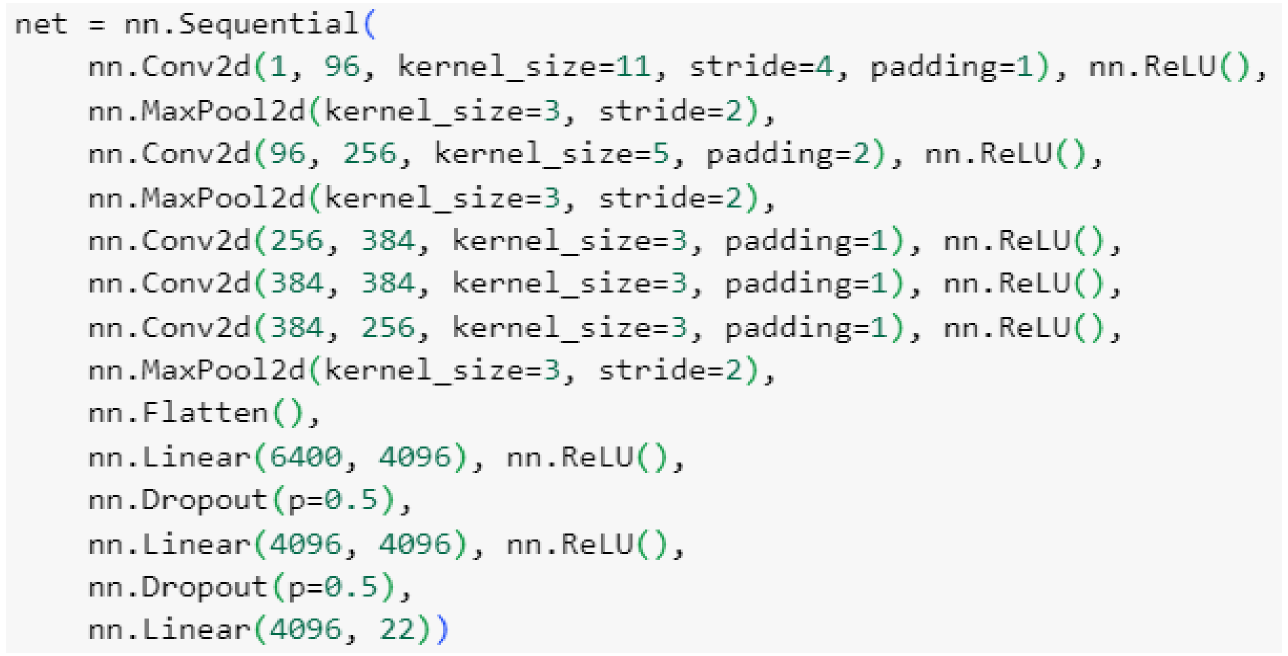
True: tricep dips
Predicted: lat pulldown

The overall **test accuracy** is 83.8%



True \ Predicted	barbell biceps curl	bench press	chest fly machine	deadlift	decline bench press	hammer curl	hip thrust	incline bench press	lat pulldown	lateral raises	leg extension	leg raises	plank	pull up	push up	romanian deadlift	russian twist	shoulder press	squat	t bar row	tricep dips	tricep pushdown
barbell biceps curl	78	1	0	0	0	0	0	1	1	3	0	0	1	0	0	0	0	1	0	2	0	0
bench press	1	40	0	0	0	0	0	1	0	0	0	1	0	0	2	0	0	0	1	0	0	0
chest fly machine	1	1	43	0	0	7	0	0	0	0	1	2	0	0	0	0	1	0	0	1	0	1
deadlift	0	0	0	46	0	0	1	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0
decline bench press	0	1	0	0	44	0	0	2	1	0	0	0	1	0	3	0	0	0	0	0	1	0
hammer curl	1	0	0	0	0	38	0	1	1	1	0	0	0	0	0	0	0	0	1	0	1	1
hip thrust	0	2	0	0	0	1	49	1	0	0	0	0	0	0	1	0	1	2	0	0	1	0
incline bench press	1	3	0	0	1	0	0	63	1	0	0	0	1	0	0	0	0	0	0	0	5	1
lat pulldown	1	0	0	0	0	0	0	0	57	0	0	0	1	0	0	0	0	0	1	0	2	1
lateral raises	2	0	0	0	1	1	0	0	0	56	0	0	1	1	0	0	0	0	0	0	1	0
leg extension	0	0	0	0	0	0	0	0	0	0	43	1	0	1	0	0	2	0	1	2	0	0
leg raises	0	4	0	0	0	0	0	0	0	0	0	42	1	0	1	0	1	0	0	0	1	1
plank	0	0	0	0	0	0	1	1	0	0	0	0	83	0	1	1	0	0	0	0	0	0
pull up	0	2	1	1	0	1	0	2	2	0	1	1	0	52	0	0	0	0	3	1	6	0
push up	0	0	0	0	0	0	0	1	0	1	1	0	3	1	47	0	2	0	0	0	1	0
romanian deadlift	0	4	0	0	1	0	0	1	0	0	0	0	1	0	0	47	0	0	0	0	0	0
russian twist	0	2	0	1	0	1	0	1	0	1	1	0	1	0	2	2	30	1	0	1	0	0
shoulder press	3	1	0	0	0	9	0	0	0	0	0	0	0	0	1	0	0	44	0	4	1	3
squat	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	1	56	0	0	0
t bar row	1	2	0	0	0	0	0	0	1	0	0	0	1	0	0	1	2	1	2	65	2	1
tricep dips	1	0	0	0	0	0	0	3	0	0	0	0	0	2	0	0	0	0	1	0	60	1
tricep pushdown	1	1	0	0	0	5	0	0	1	0	1	0	0	0	3	0	0	1	0	0	0	57

Second architecture: AlexNet



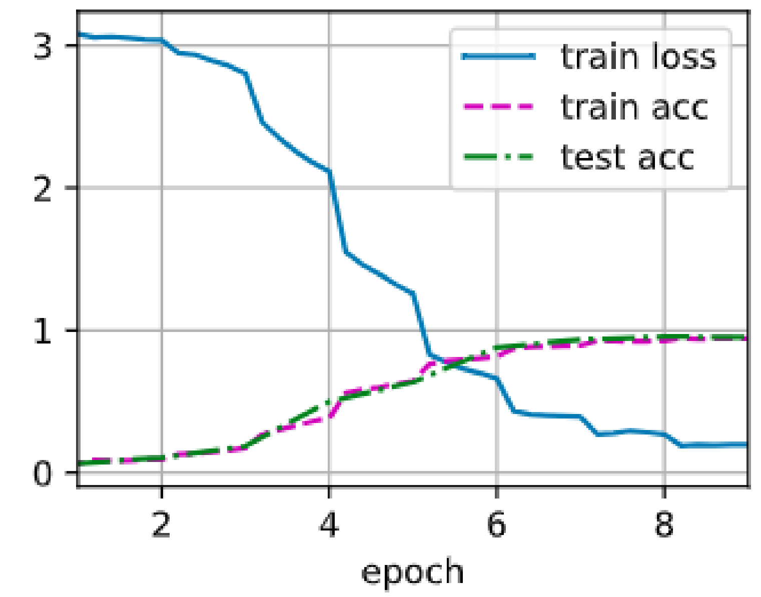
```
net = nn.Sequential(
    nn.Conv2d(1, 96, kernel_size=11, stride=4, padding=1), nn.ReLU(),
    nn.MaxPool2d(kernel_size=3, stride=2),
    nn.Conv2d(96, 256, kernel_size=5, padding=2), nn.ReLU(),
    nn.MaxPool2d(kernel_size=3, stride=2),
    nn.Conv2d(256, 384, kernel_size=3, padding=1), nn.ReLU(),
    nn.Conv2d(384, 384, kernel_size=3, padding=1), nn.ReLU(),
    nn.Conv2d(384, 256, kernel_size=3, padding=1), nn.ReLU(),
    nn.MaxPool2d(kernel_size=3, stride=2),
    nn.Flatten(),
    nn.Linear(6400, 4096), nn.ReLU(),
    nn.Dropout(p=0.5),
    nn.Linear(4096, 4096), nn.ReLU(),
    nn.Dropout(p=0.5),
    nn.Linear(4096, 22))
```

The architecture is composed by:

- A 11x11 convolution layer 1, followed by a max pooling
- A 5x5 convolutional layer 2, followed by a max pooling
- A 3x3 convolutional layer 3
- A 3x3 convolutional layer 4
- A 3x3 convolutional layer 5, followed by a max pooling
- A flatten layer
- A fully connected layer 1
- A Dropout layer 1
- A fully connected layer 2
- A dropout layer 2
- A fully connected layer 3

```
loss 0.196, train acc 0.944, val acc 0.951
721.9 examples/sec on cuda:0
```

Model result:



AlexNet Predictions

Some examples:

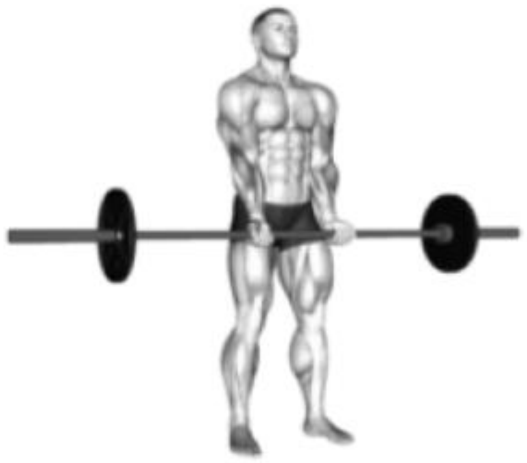
True: pull up
Predicted: pull up

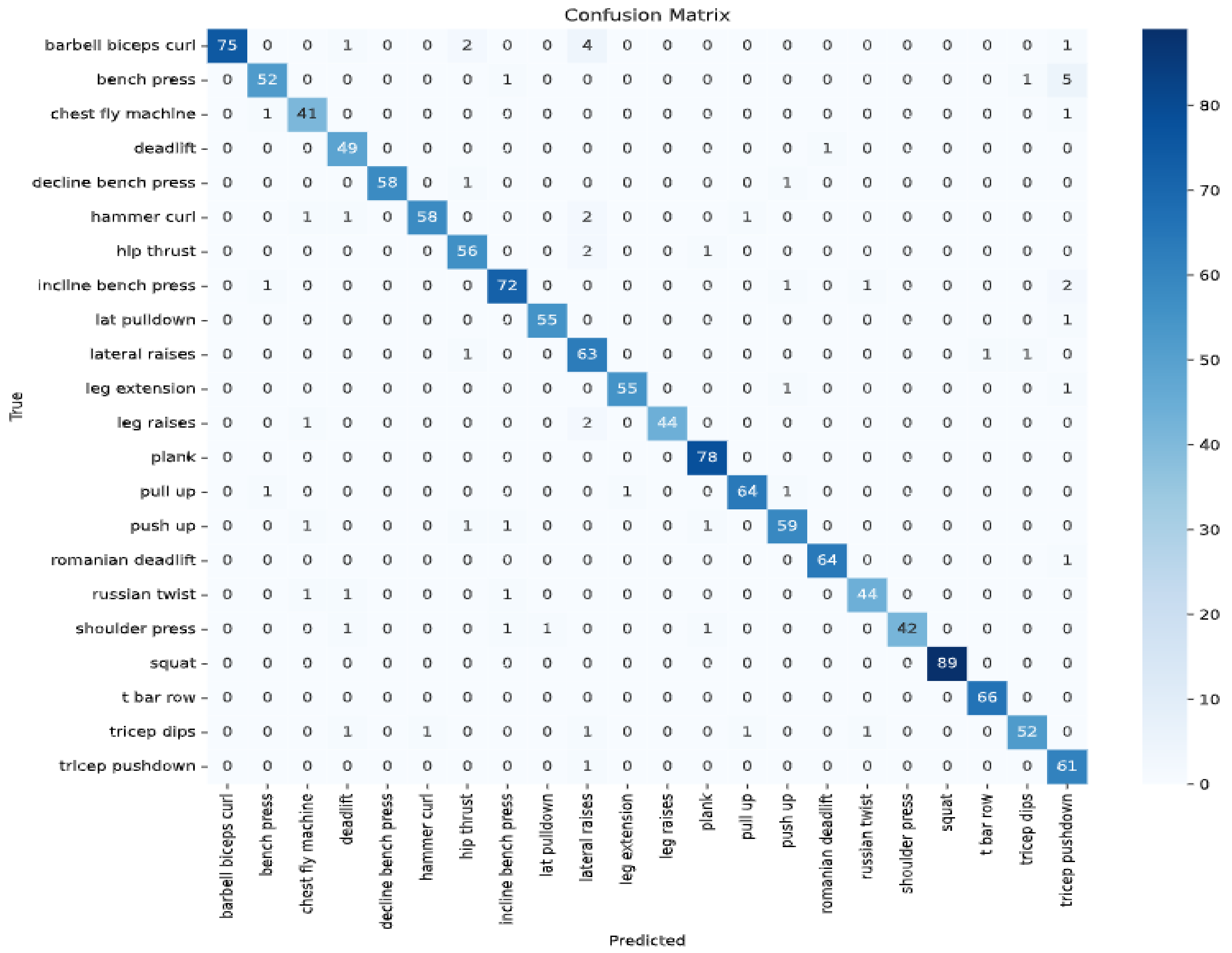
True: romanian deadlift
Predicted: romanian deadlift

True: bench press
Predicted: bench press

We created a function to display *missclassified images*:

True: barbell biceps curl
Predicted: lateral raises





True \ Predicted	barbell biceps curl	bench press	chest fly machine	deadlift	decline bench press	hammer curl	hip thrust	incline bench press	lat pulldown	lateral raises	leg extension	leg raises	plank	pull up	push up	romanian deadlift	russian twist	shoulder press	squat	t bar row	tricep dips	tricep pushdown
barbell biceps curl	75	0	0	1	0	0	2	0	0	4	0	0	0	0	0	0	0	0	0	0	0	1
bench press	0	52	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	5
chest fly machine	0	1	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
deadlift	0	0	0	49	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
decline bench press	0	0	0	0	58	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
hammer curl	0	0	1	1	0	58	0	0	0	2	0	0	0	1	0	0	0	0	0	0	0	0
hip thrust	0	0	0	0	0	0	56	0	0	2	0	0	1	0	0	0	0	0	0	0	0	0
incline bench press	0	1	0	0	0	0	0	72	0	0	0	0	0	0	1	0	1	0	0	0	0	2
lat pulldown	0	0	0	0	0	0	0	0	55	0	0	0	0	0	0	0	0	0	0	0	0	1
lateral raises	0	0	0	0	0	0	1	0	0	63	0	0	0	0	0	0	0	0	0	1	1	0
leg extension	0	0	0	0	0	0	0	0	0	0	55	0	0	0	1	0	0	0	0	0	0	1
leg raises	0	0	1	0	0	0	0	0	0	2	0	44	0	0	0	0	0	0	0	0	0	0
plank	0	0	0	0	0	0	0	0	0	0	0	0	78	0	0	0	0	0	0	0	0	0
pull up	0	1	0	0	0	0	0	0	0	0	1	0	0	64	1	0	0	0	0	0	0	0
push up	0	0	1	0	0	0	1	1	0	0	0	0	1	0	59	0	0	0	0	0	0	0
romanian deadlift	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	64	0	0	0	0	0	1
russian twist	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	44	0	0	0	0	0
shoulder press	0	0	0	1	0	0	0	1	1	0	0	0	1	0	0	0	0	42	0	0	0	0
squat	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	89	0	0	0
t bar row	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	66	0	0
tricep dips	0	0	0	1	0	1	0	0	0	1	0	0	0	1	0	0	1	0	0	0	52	0
tricep pushdown	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	61

The overall **Test Accuracy** Is 95.39%

Conclusions

	LeNet	AlexNet
Epochs	9	9
Training loss	0.642	0.196
Training accuracy	82.3%	94.4%
Test accuracy	83.8%	95.39%
Examples per seconds	1071.7	721.9

Thank you for your attention!