

```
%reload_ext autoreload
%autoreload 2
%matplotlib inline

from fastai.vision import *

bs = 64

path = untar_data(URLs.PETS)/'images'

↳ Downloading https://s3.amazonaws.com/fast-ai-imageclas/oxford-iiit-pet

tfms = get_transforms(max_rotate=20, max_zoom=1.3, max_lighting=0.4, max_warp=0.4,
                      p_affine=1., p_lighting=1.)

src = ImageList.from_folder(path).split_by_rand_pct(0.2, seed=2)

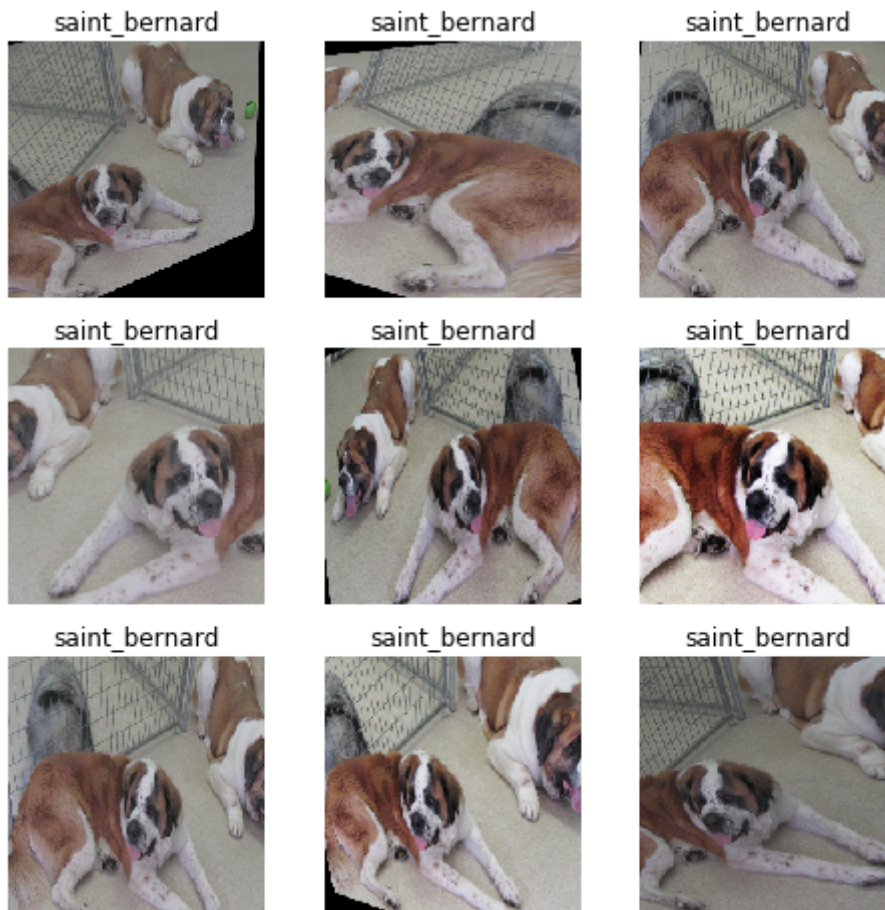
def get_data(size, bs, padding_mode='reflection'):
    return (src.label_from_re(r'([^\s/]+)_\d+.jpg$')
            .transform(tfms, size=size, padding_mode=padding_mode)
            .databunch(bs=bs).normalize(imagenet_stats))

data = get_data(224, bs, 'zeros')

def _plot(i,j,ax):
    x,y = data.train_ds[3]
    x.show(ax, y=y)

plot_multi(_plot, 3, 3, figsize=(8,8))

↳
```



```
data = get_data(224,bs)
```

```
plot_multi(_plot, 3, 3, figsize=(8,8))
```





```
gc.collect()
learn = cnn_learner(data, models.resnet34, metrics=error_rate, bn_final=True)
```

```
↳ Downloading: "https://download.pytorch.org/models/resnet34-333f7ec4.pth" to /root/.cache/torch/hub/
100%|██████████| 83.3M/83.3M [00:00<00:00, 325MB/s]
```

```
learn.fit_one_cycle(3, slice(1e-2), pct_start=0.8)
```

```
↳
```

epoch	train_loss	valid_loss	error_rate	time
0	1.644152	0.381082	0.104195	01:30
1	0.959373	0.290972	0.086604	01:30
2	0.711118	0.260182	0.076455	01:31

```
learn.unfreeze()
learn.fit_one_cycle(2, max_lr=slice(1e-6,1e-3), pct_start=0.8)
```

```
↳
```

epoch	train_loss	valid_loss	error_rate	time
0	0.577004	0.249711	0.063599	01:33
1	0.534328	0.233516	0.064953	01:34

```
data = get_data(352,bs)
learn.data = data
```

```
learn.fit_one_cycle(2, max_lr=slice(1e-6,1e-4))
```

```
↳
```

epoch	train_loss	valid_loss	error_rate	time
0	0.504815	0.229825	0.062246	02:45
1	0.470520	0.232784	0.060893	02:43

```
learn.save('352')
```

```
data = get_data(352,16)
```

```
learn = cnn_learner(data, models.resnet34, metrics=error_rate, bn_final=True).load('352')
```

```
idx=0
```

```
x,y = data.valid_ds[idx]
```

```
x.show()
```

```
data.valid_ds.y[idx]
```

```
↳ Category British_Shorthair
```



```
k = tensor([
  [0.  , -5/3, 1],
  [-5/3, -5/3, 1],
  [1.  , 1   , 1],
]).expand(1,3,3,3)/6
```

```
k
```

```
↳ tensor([[[[ 0.0000, -0.2778,  0.1667],
              [-0.2778, -0.2778,  0.1667],
              [ 0.1667,  0.1667,  0.1667]],
            [[ 0.0000, -0.2778,  0.1667],
              [-0.2778, -0.2778,  0.1667],
              [ 0.1667,  0.1667,  0.1667]],
            [[ 0.0000, -0.2778,  0.1667],
              [-0.2778, -0.2778,  0.1667],
              [ 0.1667,  0.1667,  0.1667]]]])
```

```
k.shape
```

```
↳ torch.Size([1, 3, 3, 3])
```

```
t = data.valid_ds[0][0].data; t.shape
```

```
↳ torch.Size([3, 352, 352])
```

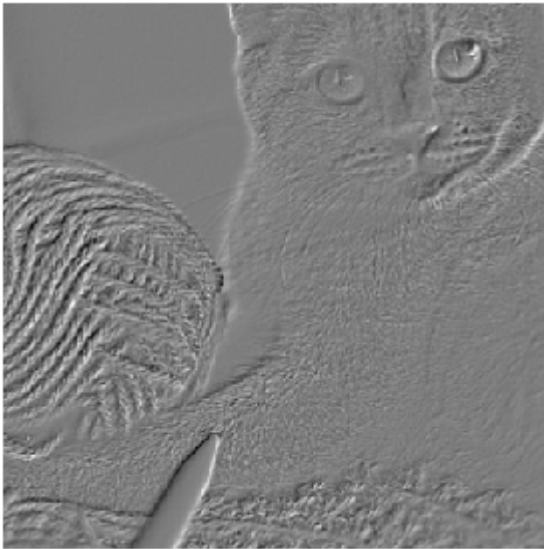
```
t[None].shape
```

```
↳ torch.Size([1, 3, 352, 352])
```

```
edge = F.conv2d(t[None], k)
```

```
show_image(edge[0], figsize=(5,5));
```

```
↳
```



```
data.c
```

```
↳ 37
```

```
learn.model
```

```
↳
```

```

(2): BasicBlock(
  (conv1): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
  (relu): ReLU(inplace=True)
  (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
)
(3): BasicBlock(
  (conv1): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
  (relu): ReLU(inplace=True)
  (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
)
(4): BasicBlock(
  (conv1): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
  (relu): ReLU(inplace=True)
  (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
)
(5): BasicBlock(
  (conv1): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
  (relu): ReLU(inplace=True)
  (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
  (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_s
)
)
(7): Sequential(
  (0): BasicBlock(
    (conv1): Conv2d(256, 512, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
    (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
    (relu): ReLU(inplace=True)
    (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
    (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
    (downsample): Sequential(
      (0): Conv2d(256, 512, kernel_size=(1, 1), stride=(2, 2), bias=False)
      (1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
    )
  )
  (1): BasicBlock(
    (conv1): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
    (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
    (relu): ReLU(inplace=True)
    (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
    (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
  )
  (2): BasicBlock(
    (conv1): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
    (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
    (relu): ReLU(inplace=True)
    (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
    (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True, track_running_s
  )
)
)
(1): Sequential(
  (0): AdaptiveConcatPool2d(
    (ap): AdaptiveAvgPool2d(output_size=1)
    (mp): AdaptiveMaxPool2d(output_size=1)
  )
)

```

```
,  
(1): Flatten()  
(2): BatchNorm1d(1024, eps=1e-05, momentum=0.1, affine=True, track_running_stats=  
(3): Dropout(p=0.25, inplace=False)  
(4): Linear(in_features=1024, out_features=512, bias=True)  
(5): ReLU(inplace=True)  
(6): BatchNorm1d(512, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T  
(7): Dropout(p=0.5, inplace=False)  
(8): Linear(in_features=512, out_features=37, bias=True)  
(9): BatchNorm1d(37, eps=1e-05, momentum=0.01, affine=True, track_running_stats=T  
)  
)
```

```
print(learn.summary())
```



Sequential

=====			
Layer (type)	Output Shape	Param #	Trainable
=====			
Conv2d	[64, 176, 176]	9,408	False
BatchNorm2d	[64, 176, 176]	128	True
ReLU	[64, 176, 176]	0	False
MaxPool2d	[64, 88, 88]	0	False
Conv2d	[64, 88, 88]	36,864	False
BatchNorm2d	[64, 88, 88]	128	True
ReLU	[64, 88, 88]	0	False
Conv2d	[64, 88, 88]	36,864	False
BatchNorm2d	[64, 88, 88]	128	True
Conv2d	[64, 88, 88]	36,864	False
BatchNorm2d	[64, 88, 88]	128	True
ReLU	[64, 88, 88]	0	False
Conv2d	[64, 88, 88]	36,864	False
BatchNorm2d	[64, 88, 88]	128	True
Conv2d	[64, 88, 88]	36,864	False
BatchNorm2d	[64, 88, 88]	128	True
ReLU	[64, 88, 88]	0	False
Conv2d	[64, 88, 88]	36,864	False
BatchNorm2d	[64, 88, 88]	128	True
Conv2d	[128, 44, 44]	73,728	False
BatchNorm2d	[128, 44, 44]	256	True
ReLU	[128, 44, 44]	0	False
Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
Conv2d	[128, 44, 44]	8,192	False
BatchNorm2d	[128, 44, 44]	256	True
Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
ReLU	[128, 44, 44]	0	False

Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
ReLU	[128, 44, 44]	0	False
Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
ReLU	[128, 44, 44]	0	False
Conv2d	[128, 44, 44]	147,456	False
BatchNorm2d	[128, 44, 44]	256	True
Conv2d	[256, 22, 22]	294,912	False
BatchNorm2d	[256, 22, 22]	512	True
ReLU	[256, 22, 22]	0	False
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
Conv2d	[256, 22, 22]	32,768	False
BatchNorm2d	[256, 22, 22]	512	True
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
ReLU	[256, 22, 22]	0	False
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
ReLU	[256, 22, 22]	0	False
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
ReLU	[256, 22, 22]	0	False
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True
Conv2d	[256, 22, 22]	589,824	False
BatchNorm2d	[256, 22, 22]	512	True