Three modifications are made for 6 channels:

1. In code/data.py, I add a new drawing function:

```
24
25
26  def drawing_to_image_with_color_aug_6channel(drawing, H, W, seq):
27
28      point=[]
29      time =[]
30      for t,(x,y) in enumerate(drawing):
31          point.append(np.array((x,y),np.float32).T)
32          time.append(np.full(len(x),t))
33
34      point = np.concatenate(point).astype(np.float32)
35      time  = np.concatenate(time ).astype(np.int32)
```

2. I add a new file training file "*code/train/train-6channel.ipynb*", and utilize the new drawing function:

```
]: def valid_augment(drawing, label, index):
#     image = drawing_to_image_with_color_v2(drawing, 96, 96)
    seq = iaa.Sequential([
    iaa.Crop(percent=(0.05, 0.05, \
                      0.05, 0.05), keep_size=True)
    ])
    image = drawing_to_image_with_color_aug_6channel(drawing, 96, 96, seq)
#     output_name = str(index) + '.png'
#     cv2.imwrite('../split/valid_img/' + output_name,image)
    return image, label, None


def train_augment(drawing, label, index):
    up_rand = np.random.random()
    right_rand = np.random.random()
    percent_crop = 0.1
    seq = iaa.Sequential([
    iaa.Fliplr(0.5),
    iaa.Crop(percent=(up_rand*percent_crop, right_rand*percent_crop, \
                      (1-up_rand)*percent_crop, (1-right_rand)*percent_crop), ke
    ])

    image = drawing_to_image_with_color_aug_6channel(drawing, 96, 96, seq)
#     image = drawing_to_image_with_color_v2(drawing, 96, 96)
    return image, label, None
```

   In predicting part, don't forget to change this place too (in def test_augment …) if you wanna use 6 channels.

3. I add a new model function "*model_seresnext50_6channel.py*"

Therefore, please use "*train-6channel.ipynb*" to train, if you wish to fine-tune your checkpoint come from 3 channels, like:

```
initial_checkpoint = \
        '../../output/backup/887_crop.pth'
```

Please set "*from_3channel*" parameter to True, like:

```
if initial_checkpoint is not None:
    log.write('\tinitial_checkpoint = %s\n' % initial_checkpoint)
    net.load_pretrain\
    (initial_checkpoint,from_3channel=True)
```

For further fine-tune your 6 channels' checkpoint, obviously, this parameter should be False.