Project 3 - Main Script

Code

Guanren Wang

2019-3-27

This file organizes **all computational steps** for evaluating baseline image processing framework. For computational steps of improved model, please see python documents.

The baseline model(GBM) is computed via Google Cloud VM with machine type: **n1-standard-16 (16 vCPUs, 60 GB memory)** and CPU platform: **Intel Broadwell**. In view of heavy computing workload, I recommend you use VM to reproduce.

Load libraries

Hide

```
if(!require("EBImage")){
  source("https://bioconductor.org/biocLite.R")
  biocLite("EBImage")
  library("EBImage")
}

if(!require("gbm")){
  install.packages("gbm")
  library("gbm")
}

library("EBImage")
library("gbm")
```

Parallel Computing Setup

Hide

```
if(!require("doParallel")){
  install.packages("doParallel")
  library("doParallel")
}

# Real physical cores in the computer
cores <- detectCores()

if(cores>1){

  if(.Platform$OS.type=="windows"){
    cl <- makeCluster(cores)
    registerDoParallel(cl, cores=cores)
  }
  else{
    if(!require("doMC")){
      install.packages("doMC")
      library("doMC")
    }
    registerDoMC(cores)
  }

  run.parallel=TRUE

}else
  run.parallel=FALSE
cores<-cores
```

Step 0: Specify Directories.

Provide directories for training images. **Low-resolution (LR)** image set and **High-resolution (HR)** image set will be in different subfolders.

Hide

```
set.seed(123)
# use relative path for reproducibility

train_dir <- "../data/train_set/" # This will be modified for different data sets.
train_LR_dir <- paste(train_dir, "LR/", sep="")
train_HR_dir <- paste(train_dir, "HR/", sep="")
train_label_path <- paste(train_dir, "label.csv", sep="")
```

Step 1: Set Up Controls For Evaluation Experiments.

In this chunk, we have a set of controls for the evaluation experiments.

- (T/F) cross-validation on the training set
- (number) K, the number of CV folds
- (T/F) whether run training

- (T/F) process features for training set
- (T/F) run evaluation on an independent test set
- (T/F) process features for test set

Hide

```
run.cv=T # run cross-validation on the training set
K <- 5  # number of CV folds
run.train=TRUE # whether run training
run.feature.train=TRUE # process features for training set
run.test=TRUE # run evaluation on an independent test set
run.feature.test=TRUE # process features for test set
```

Using cross-validation or independent test set evaluation, we compare the performance of models with different specifications. In this project, we use GBM with different `depth`. In the following chunk, we list, in a vector, setups (in this case, `depth`) corresponding to models that we will compare. In your project, you might compare very different classifiers. You can assign them numerical IDs and labels specific to your project.

Hide

```
model_values <- 2:6
model_labels = paste("GBM with depth =", model_values)
```

Step 2: Import Training Images Class Labels.

We provide extra information of image label: car (0), flower (1), market (2). These labels are not necessary for your model.

Hide

```
extra_label <- read.csv(train_label_path, colClasses=c("NULL", NA, NA))
```

Step 3: Construct Features and Responses

`feature.R` should be the wrapper for all your feature engineering functions and options. The function `feature( )` should have options that correspond to different scenarios for your project and produces an R object that contains features and responses that are required by all the models you are going to evaluate later.

`feature.R`:

- Input: a path for low-resolution images.
- Input: a path for high-resolution images.
- Output: an RData file that contains extracted features and corresponding responses

I do recommend using non-parallel version here, as extract feature is a data-intensive — not computation intensive step. Thus, parallel computing won't significantly save time — even increase running time, for the parallel computing will raise additional coordination cost, depends on different hardwares:

Running Time	HPC	Pixelbook	old laptop
Parallel	19	55	80

Running Time	HPC	Pixelbook	old laptop
Non-Parallel	51	33	90

Hide

```
if(run.parallel)
  source("../lib/feature_parallel.R") else
    source("../lib/feature.R")
```

Hide

```
tm_feature_train <- NA
if(run.feature.train){
  tm0=proc.time()
  dat_train <- feature(train_LR_dir, train_HR_dir)
  tm_feature_train=proc.time()-tm0

  save(dat_train, file="../output/feature_train.RData")
}else{
  load("../output/feature_train.RData")
}
feat_train <- dat_train$feature
label_train <- dat_train$label
rm(dat_train)
tmp=gc() # release memory
```

Step 4: Train a classification model with training images

Call the train model and test model from library.

`train.R` and `test.R` should be wrappers for all your model training steps and your classification/prediction steps.

`train.R`:

- Input: a path that points to the training set features and responses.
- Output: an RData file that contains trained classifiers in the forms of R objects: models/settings/links to external trained configurations.

`test.R`:

- Input: a path that points to the test set features.
- Input: an R object that contains a trained classifier.
- Output: an R object of response predictions on the test set. If there are multiple classifiers under evaluation, there should be multiple sets of label predictions.

Hide

```
if(run.parallel)
  source("../lib/train_parallel.R") else
    source("../lib/train.R")

### The test parallel will increase much more time, so we use non-paralleled one
source("../lib/test.R")
```

Model selection with cross-validation

- Do model selection by choosing among different values of training model parameters, that is, the interaction depth for GBM in this example.

Hide

```
source("../lib/cross_validation.R")
if(run.cv){
  err_cv <- array(dim=c(length(model_values), 2))
  for(k in 1:length(model_values)){
    cat("k=", k, "\n")
    err_cv[k,] <- cv.function(feat_train, label_train, model_values[k], K)
    system("free -m")
  }
  save(err_cv, file="../output/err_cv.RData")
}else{
  load("../output/err_cv.RData")
}
tmp=gc() # release memory
```

Visualize cross-validation results

Hide

```
plot(model_values, err_cv[,1], xlab="Interaction Depth", ylab="CV Error",
       main="Cross Validation Error", type="n", ylim=c(0, 0.01))
points(model_values, err_cv[,1], col="blue", pch=16)
```

Hide

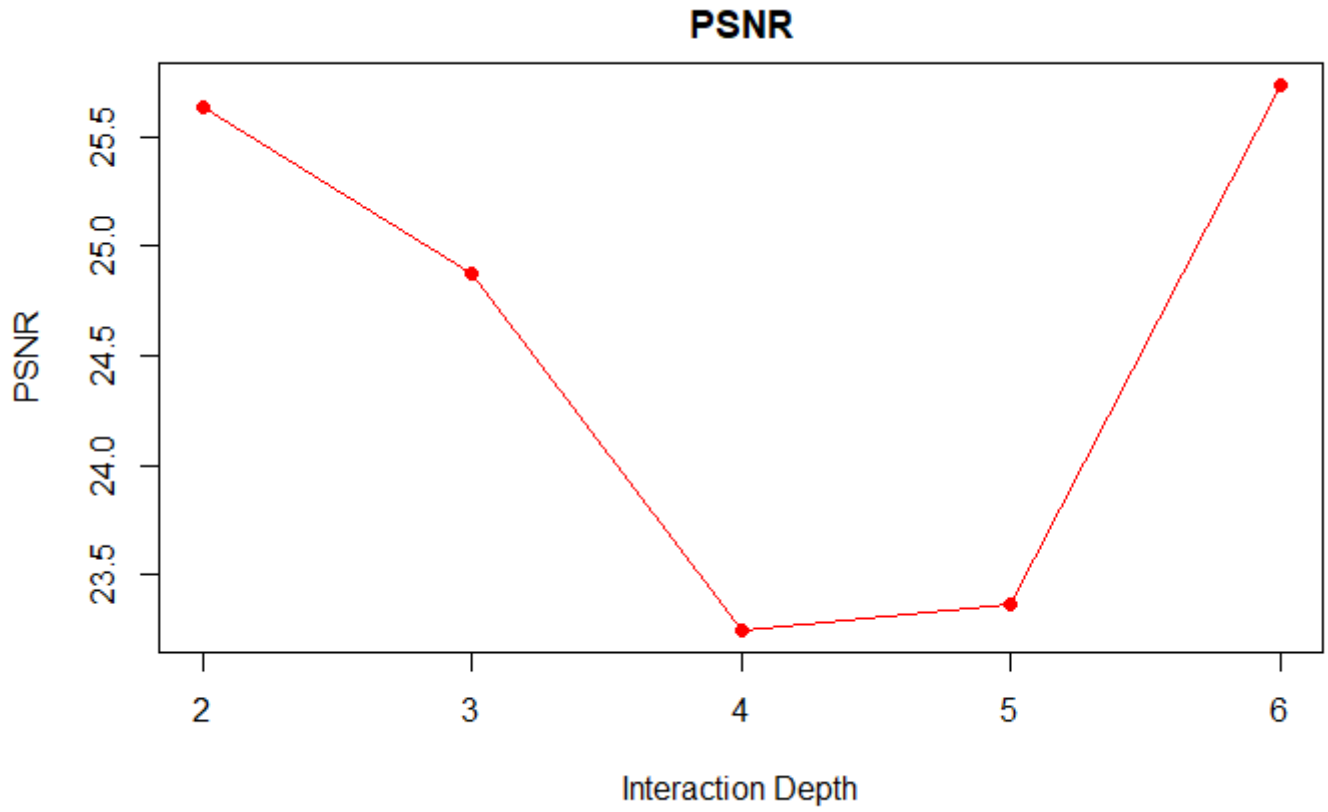
```
lines(model_values, err_cv[,1], col="blue")
arrows(model_values, err_cv[,1]-err_cv[,2], model_values, err_cv[,1]+err_cv[,2],
        length=0.1, angle=90, code=3)
```

Hide

```
psnr<-10*log10(1/err_cv[,1])
plot(model_values, psnr, xlab="Interaction Depth", ylab="PSNR",
       main="PSNR", type="n")
points(model_values, psnr, col="red", pch=16)
```

Hide

```
lines(model_values, psnr, col="red")
```



Choose the "best" parameter value

Hide

```
model_best=model_values[1]

model_best <- model_values[which.min(err_cv[,1])]
cat("The min cv is",min(err_cv[,1]),"for depth",model_best,"\n")

par_best <- list(depth=model_best)
```

Train the model with the entire training set using the selected model (model parameter) via cross-validation.

Hide

```
source("../lib/train_parallel.R")
if(run.train){
  tm0=proc.time()
  fit_train <- train(feat_train, label_train, par_best)
  tm_train=proc.time()-tm0
  save(fit_train, file="../output/fit_train.RData")
}else{
  load("../output/fit_train.RData")
}
tmp=gc()
```

Step 5: Super-resolution for test images

Feed the final training model with the completely holdout testing data. + `superResolution.R` + Input: a path that points to the folder of low-resolution test images. + Input: a path that points to the folder (empty) of high-resolution test images. + Input: an R object that contains tuned predictors. + Output: construct high-resolution versions for each low-resolution test image.

Hide

```
# useually we use parallel computing to generate new images, but because of incompatibil
ity of operating systems, superrevolution doesn't work, in this case we could do this wi
thout parallel computing.
# for non-parallel computing, just delete pound sign below

# run.parallel=F
if(run.parallel)
  source("../lib/superResolution_parallel.R") else
    source("../lib/superResolution.R")
test_dir <- "../data/test_set/" # This will be modified for different data sets.
test_LR_dir <- paste(train_dir, "LR/", sep="")
test_HR_dir <- paste(test_dir, "HR/", sep="")

if(run.test){
  tm0=proc.time()
  superResolution(test_LR_dir, test_HR_dir, fit_train)
  tm_test=proc.time()-tm0
}
run.parallel=T
```

step 6: Summarize Running Time

Prediction performance matters, so does the running times for constructing features and for training the model, especially when the computation resource is limited.

Hide

```
cat("Time for constructing training features=", tm_feature_train[3], "s \n")
cat("Time for training model=", tm_train[3], "s \n")
cat("Time for super-resolution=", tm_test[3], "s \n")
```

step 7: Calculate MSE and PSNR

Hide

```
train_dir <- "../data/train_set/HR/"
test_dir <- "../data/test_set/HR/"
source("../lib/Evaluation.R")
mp=msepsnr(train_dir, test_dir)
cat("MSE is", mp[1],'\n')

cat("PSNR is", mp[2])
```

step 8: Stop Parallel Computing

Hide

```
if(run.parallel & .Platform$OS.type=="windows"){
  stopImplicitCluster()
  stopCluster(cl)
}
```

Summary:

Baseline Model

Our baseline model consists of four **steps**:
1. Preprocessing and Feature Extraction
2. Model training, evaluation(parameter tuning) and selection
3. Making prediction using selected model
4. Evaluating prediction and running time of model

Results:

We ran feature extraction by sampling 1000 points from the LR image pixels and takin the 8 neighboring pixels for each point as features.

Optimal depth is 6 with cross-validation error 0.00266912

Time for constructing training features= 92.81 s Time for training model= 379.08 s Time for super-resolution= 2717.55 s (1500 images)

MSE is 0.002776163
PSNR is 27.41872.

It is better than the nerest-neighbor method which uses the value of nearby translated pixel values for the output pixel values. MSE of this method is 0.004035 and PSNR is 25.3926.

It is also better than bilinear interpolation which uses the weighted average of two translated pixel values for each output pixel value. The MSE is 0.003277 and PSNR is 26.3723.

Additionally, it is better than bicubic interpolation as well, which uses the weighted average of four translated pixel values for each output pixel value. MSE is 0.004927 and the PSNR is 24.1553.

SRGAN

In [1]:
```
import os, sys
sys.path.append('../lib/SRGAN/')
import tensorflow as tf
from main import train, predict
```

Train

In [2]:
```
#tf.reset_default_graph()
# We can do validation!
#train(train_lr_path, train_hr_path, save_path, validation=True, ratio=
0.9, n_epoch_init=20, n_epoch=20)

##### Parameters #####:
## data and result path
train_lr_path = '../data/train_set/LR'
train_hr_path = '../data/train_set/HR'
save_path = '../output/SRGAN2'
save_every_epoch = 1
# validation: whether to split data into train set and validation setbn7
## Adam
batch_size=32
# lr_init=1e-4
# beta1=0.9
## Initialize generator
# n_epoch_init
## train adversial net
# n_epoch
# lr_decay=0.1
train(train_lr_path=train_lr_path, train_hr_path=train_hr_path, save_pat
h=save_path,
      save_every_epoch=save_every_epoch, validation=False, batch_size=ba
tch_size, n_epoch_init=10, n_epoch=20)
```

```
[TL] [!] ../output/SRGAN2/srgan_ginit exists ...
[TL] [!] ../output/SRGAN2/srgan_gan exists ...
[TL] [!] ../output/SRGAN2/checkpoint exists ...
[TL] read 32 from ../data/train_set/LR
[TL] read 64 from ../data/train_set/LR
[TL] read 96 from ../data/train_set/LR
[TL] read 128 from ../data/train_set/LR
[TL] read 160 from ../data/train_set/LR
[TL] read 192 from ../data/train_set/LR
[TL] read 224 from ../data/train_set/LR
[TL] read 256 from ../data/train_set/LR
[TL] read 288 from ../data/train_set/LR
[TL] read 320 from ../data/train_set/LR
[TL] read 352 from ../data/train_set/LR
[TL] read 384 from ../data/train_set/LR
[TL] read 416 from ../data/train_set/LR
[TL] read 448 from ../data/train_set/LR
[TL] read 480 from ../data/train_set/LR
[TL] read 512 from ../data/train_set/LR
[TL] read 544 from ../data/train_set/LR
[TL] read 576 from ../data/train_set/LR
[TL] read 608 from ../data/train_set/LR
[TL] read 640 from ../data/train_set/LR
[TL] read 672 from ../data/train_set/LR
[TL] read 704 from ../data/train_set/LR
[TL] read 736 from ../data/train_set/LR
[TL] read 768 from ../data/train_set/LR
[TL] read 800 from ../data/train_set/LR
[TL] read 832 from ../data/train_set/LR
[TL] read 864 from ../data/train_set/LR
[TL] read 896 from ../data/train_set/LR
[TL] read 928 from ../data/train_set/LR
[TL] read 960 from ../data/train_set/LR
[TL] read 992 from ../data/train_set/LR
[TL] read 1024 from ../data/train_set/LR
[TL] read 1056 from ../data/train_set/LR
[TL] read 1088 from ../data/train_set/LR
[TL] read 1120 from ../data/train_set/LR
[TL] read 1152 from ../data/train_set/LR
[TL] read 1184 from ../data/train_set/LR
[TL] read 1216 from ../data/train_set/LR
[TL] read 1248 from ../data/train_set/LR
[TL] read 1280 from ../data/train_set/LR
[TL] read 1312 from ../data/train_set/LR
[TL] read 1344 from ../data/train_set/LR
[TL] read 1376 from ../data/train_set/LR
[TL] read 1408 from ../data/train_set/LR
[TL] read 1440 from ../data/train_set/LR
[TL] read 1472 from ../data/train_set/LR
[TL] read 1500 from ../data/train_set/LR
[TL] read 32 from ../data/train_set/HR
[TL] read 64 from ../data/train_set/HR
[TL] read 96 from ../data/train_set/HR
[TL] read 128 from ../data/train_set/HR
[TL] read 160 from ../data/train_set/HR
[TL] read 192 from ../data/train_set/HR
[TL] read 224 from ../data/train_set/HR
```

```
[TL] read 256 from ../data/train_set/HR
[TL] read 288 from ../data/train_set/HR
[TL] read 320 from ../data/train_set/HR
[TL] read 352 from ../data/train_set/HR
[TL] read 384 from ../data/train_set/HR
[TL] read 416 from ../data/train_set/HR
[TL] read 448 from ../data/train_set/HR
[TL] read 480 from ../data/train_set/HR
[TL] read 512 from ../data/train_set/HR
[TL] read 544 from ../data/train_set/HR
[TL] read 576 from ../data/train_set/HR
[TL] read 608 from ../data/train_set/HR
[TL] read 640 from ../data/train_set/HR
[TL] read 672 from ../data/train_set/HR
[TL] read 704 from ../data/train_set/HR
[TL] read 736 from ../data/train_set/HR
[TL] read 768 from ../data/train_set/HR
[TL] read 800 from ../data/train_set/HR
[TL] read 832 from ../data/train_set/HR
[TL] read 864 from ../data/train_set/HR
[TL] read 896 from ../data/train_set/HR
[TL] read 928 from ../data/train_set/HR
[TL] read 960 from ../data/train_set/HR
[TL] read 992 from ../data/train_set/HR
[TL] read 1024 from ../data/train_set/HR
[TL] read 1056 from ../data/train_set/HR
[TL] read 1088 from ../data/train_set/HR
[TL] read 1120 from ../data/train_set/HR
[TL] read 1152 from ../data/train_set/HR
[TL] read 1184 from ../data/train_set/HR
[TL] read 1216 from ../data/train_set/HR
[TL] read 1248 from ../data/train_set/HR
[TL] read 1280 from ../data/train_set/HR
[TL] read 1312 from ../data/train_set/HR
[TL] read 1344 from ../data/train_set/HR
[TL] read 1376 from ../data/train_set/HR
[TL] read 1408 from ../data/train_set/HR
[TL] read 1440 from ../data/train_set/HR
[TL] read 1472 from ../data/train_set/HR
[TL] read 1500 from ../data/train_set/HR
[TL] InputLayer  SRGAN_g/in: (?, 96, 96, 3)
[TL] Conv2d SRGAN_g/n64s1/c: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d SRGAN_g/n64s1/c1/0: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/0: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/0: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/0: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/0: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/1: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/1: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
```

```
[TL] Conv2d SRGAN_g/n64s1/c2/1: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/1: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/1: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/2: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/2: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/2: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/2: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/2: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/3: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/3: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/3: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/3: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/3: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/4: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/4: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/4: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/4: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/4: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/5: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/5: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/5: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/5: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/5: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/6: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/6: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/6: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/6: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/6: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/7: n_filter: 64 filter_size: (3, 3) stride
```

```
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/7: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/7: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/7: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/7: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/8: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/8: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/8: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/8: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/8: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/9: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/9: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/9: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/9: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/9: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/10: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/10: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/10: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/10: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/10: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/11: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/11: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/11: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/11: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/11: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/12: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/12: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/12: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/12: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: True
```

```
[TL] ElementwiseLayer SRGAN_g/b_residual_add/12: size: (?, 96, 96, 64) fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/13: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/13: decay: 0.900000 epsilon: 0.000010 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/13: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/13: decay: 0.900000 epsilon: 0.000010 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/13: size: (?, 96, 96, 64) fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/14: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/14: decay: 0.900000 epsilon: 0.000010 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/14: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/14: decay: 0.900000 epsilon: 0.000010 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/14: size: (?, 96, 96, 64) fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/15: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/15: decay: 0.900000 epsilon: 0.000010 act: relu is_train: True
[TL] Conv2d SRGAN_g/n64s1/c2/15: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/15: decay: 0.900000 epsilon: 0.000010 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/b_residual_add/15: size: (?, 96, 96, 64) fn: add
[TL] Conv2d SRGAN_g/n64s1/c/m: n_filter: 64 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b/m: decay: 0.900000 epsilon: 0.000010 act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_g/add3: size: (?, 96, 96, 64) fn: add
[TL] Conv2d SRGAN_g/n256s1/1: n_filter: 256 filter_size: (3, 3) strides: (1, 1) pad: SAME act: No Activation
[TL] SubpixelConv2d  SRGAN_g/pixelshufflerx2/1: scale: 2 n_out_channel: 64 act: relu
[TL] Conv2d SRGAN_g/out: n_filter: 3 filter_size: (1, 1) strides: (1, 1) pad: SAME act: tanh
[TL] WARNING: Function: `tensorlayer.layers.utils.set_name_reuse` (in file: /Users/james/anaconda3/lib/python3.6/site-packages/tensorlayer/layers/utils.py) is deprecated and will be removed after 2018-06-30.
Instructions for updating: TensorLayer relies on TensorFlow to check name reusing

[TL] WARNING: this method is DEPRECATED and has no effect, please remove it from your code.
[TL] InputLayer  SRGAN_d/input/images: (?, 192, 192, 3)
[TL] Conv2d SRGAN_d/h0/c: n_filter: 64 filter_size: (4, 4) strides: (2, 2) pad: SAME act: <lambda>
[TL] WARNING: Function: `tensorlayer.activation.leaky_relu` (in file: /Users/james/anaconda3/lib/python3.6/site-packages/tensorlayer/activation.py) is deprecated and will be removed after 2018-09-30.
```

```
Instructions for updating: This API is deprecated. Please use as `tf.n
n.leaky_relu`

[TL] Conv2d SRGAN_d/h1/c: n_filter: 128 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h1/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h2/c: n_filter: 256 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h2/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h3/c: n_filter: 512 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h3/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h4/c: n_filter: 1024 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h4/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h5/c: n_filter: 2048 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h5/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h6/c: n_filter: 1024 filter_size: (1, 1) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h6/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h7/c: n_filter: 512 filter_size: (1, 1) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h7/bn: decay: 0.900000 epsilon: 0.000010 ac
t: No Activation is_train: True
[TL] Conv2d SRGAN_d/res/c: n_filter: 128 filter_size: (1, 1) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/res/bn: decay: 0.900000 epsilon: 0.000010 a
ct: <lambda> is_train: True
[TL] Conv2d SRGAN_d/res/c2: n_filter: 128 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/res/bn2: decay: 0.900000 epsilon: 0.000010
act: <lambda> is_train: True
[TL] Conv2d SRGAN_d/res/c3: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/res/bn3: decay: 0.900000 epsilon: 0.000010
act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_d/res/add: size: (?, 3, 3, 512) fn: add
[TL] FlattenLayer SRGAN_d/ho/flatten: 4608
[TL] DenseLayer  SRGAN_d/ho/dense: 1 No Activation
[TL] WARNING: this method is DEPRECATED and has no effect, please remov
e it from your code.
[TL] InputLayer  SRGAN_d/input/images: (?, 192, 192, 3)
[TL] Conv2d SRGAN_d/h0/c: n_filter: 64 filter_size: (4, 4) strides: (2,
2) pad: SAME act: <lambda>
[TL] Conv2d SRGAN_d/h1/c: n_filter: 128 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h1/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h2/c: n_filter: 256 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
```

```
[TL] BatchNormLayer SRGAN_d/h2/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h3/c: n_filter: 512 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h3/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h4/c: n_filter: 1024 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h4/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h5/c: n_filter: 2048 filter_size: (4, 4) strides:
(2, 2) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h5/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h6/c: n_filter: 1024 filter_size: (1, 1) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h6/bn: decay: 0.900000 epsilon: 0.000010 ac
t: <lambda> is_train: True
[TL] Conv2d SRGAN_d/h7/c: n_filter: 512 filter_size: (1, 1) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/h7/bn: decay: 0.900000 epsilon: 0.000010 ac
t: No Activation is_train: True
[TL] Conv2d SRGAN_d/res/c: n_filter: 128 filter_size: (1, 1) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/res/bn: decay: 0.900000 epsilon: 0.000010 a
ct: <lambda> is_train: True
[TL] Conv2d SRGAN_d/res/c2: n_filter: 128 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/res/bn2: decay: 0.900000 epsilon: 0.000010
act: <lambda> is_train: True
[TL] Conv2d SRGAN_d/res/c3: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_d/res/bn3: decay: 0.900000 epsilon: 0.000010
act: No Activation is_train: True
[TL] ElementwiseLayer SRGAN_d/res/add: size: (?, 3, 3, 512) fn: add
[TL] FlattenLayer SRGAN_d/ho/flatten: 4608
[TL] DenseLayer  SRGAN_d/ho/dense: 1 No Activation
[TL] InputLayer  VGG19/input: (?, 224, 224, 3)
[TL] Conv2d VGG19/conv1_1: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv1_2: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool1: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv2_1: n_filter: 128 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv2_2: n_filter: 128 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool2: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv3_1: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv3_2: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv3_3: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv3_4: n_filter: 256 filter_size: (3, 3) strides:
```

```
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool3: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv4_1: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv4_2: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv4_3: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv4_4: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool4: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv5_1: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv5_2: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv5_3: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv5_4: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool5: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] FlattenLayer VGG19/flatten: 25088
[TL] DenseLayer  VGG19/fc6: 4096 relu
[TL] DenseLayer  VGG19/fc7: 4096 relu
[TL] DenseLayer  VGG19/fc8: 1000 No Activation
[TL] InputLayer  VGG19/input: (?, 224, 224, 3)
[TL] Conv2d VGG19/conv1_1: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv1_2: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool1: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv2_1: n_filter: 128 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv2_2: n_filter: 128 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool2: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv3_1: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv3_2: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv3_3: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv3_4: n_filter: 256 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool3: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv4_1: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv4_2: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv4_3: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv4_4: n_filter: 512 filter_size: (3, 3) strides:
```

```
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool4: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] Conv2d VGG19/conv5_1: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv5_2: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv5_3: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d VGG19/conv5_4: n_filter: 512 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] MaxPool2d VGG19/pool5: filter_size: (2, 2) strides: (2, 2) paddin
g: SAME
[TL] FlattenLayer VGG19/flatten: 25088
[TL] DenseLayer  VGG19/fc6: 4096 relu
[TL] DenseLayer  VGG19/fc7: 4096 relu
[TL] DenseLayer  VGG19/fc8: 1000 No Activation
[TL] InputLayer  SRGAN_g/in: (?, 96, 96, 3)
[TL] Conv2d SRGAN_g/n64s1/c: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d SRGAN_g/n64s1/c1/0: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/0: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/0: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/0: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/0: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/1: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/1: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/1: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/1: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/1: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/2: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/2: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/2: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/2: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/2: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/3: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/3: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/3: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/3: decay: 0.900000 epsilon: 0.0000
```

```
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/3: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/4: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/4: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/4: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/4: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/4: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/5: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/5: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/5: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/5: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/5: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/6: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/6: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/6: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/6: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/6: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/7: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/7: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/7: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/7: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/7: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/8: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/8: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/8: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/8: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/8: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/9: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/9: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
```

```
[TL] Conv2d SRGAN_g/n64s1/c2/9: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/9: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/9: size: (?, 96, 96, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/10: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/10: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/10: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/10: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/10: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/11: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/11: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/11: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/11: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/11: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/12: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/12: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/12: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/12: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/12: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/13: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/13: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/13: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/13: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/13: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/14: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/14: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/14: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/14: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/14: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c1/15: n_filter: 64 filter_size: (3, 3) strid
```

```
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/15: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/15: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/15: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/15: size: (?, 96, 96, 64)
fn: add
[TL] Conv2d SRGAN_g/n64s1/c/m: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b/m: decay: 0.900000 epsilon: 0.00001
0 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/add3: size: (?, 96, 96, 64) fn: add
[TL] Conv2d SRGAN_g/n256s1/1: n_filter: 256 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] SubpixelConv2d  SRGAN_g/pixelshufflerx2/1: scale: 2 n_out_channel:
64 act: relu
[TL] Conv2d SRGAN_g/out: n_filter: 3 filter_size: (1, 1) strides: (1,
1) pad: SAME act: tanh
[TL]   [*] geting variables with SRGAN_g
[TL]   got   0: SRGAN_g/n64s1/c/kernel:0   (3, 3, 3, 64)
[TL]   got   1: SRGAN_g/n64s1/c/bias:0   (64,)
[TL]   got   2: SRGAN_g/n64s1/c1/0/kernel:0   (3, 3, 64, 64)
[TL]   got   3: SRGAN_g/n64s1/c1/0/bias:0   (64,)
[TL]   got   4: SRGAN_g/n64s1/b1/0/beta:0   (64,)
[TL]   got   5: SRGAN_g/n64s1/b1/0/gamma:0   (64,)
[TL]   got   6: SRGAN_g/n64s1/c2/0/kernel:0   (3, 3, 64, 64)
[TL]   got   7: SRGAN_g/n64s1/c2/0/bias:0   (64,)
[TL]   got   8: SRGAN_g/n64s1/b2/0/beta:0   (64,)
[TL]   got   9: SRGAN_g/n64s1/b2/0/gamma:0   (64,)
[TL]   got  10: SRGAN_g/n64s1/c1/1/kernel:0   (3, 3, 64, 64)
[TL]   got  11: SRGAN_g/n64s1/c1/1/bias:0   (64,)
[TL]   got  12: SRGAN_g/n64s1/b1/1/beta:0   (64,)
[TL]   got  13: SRGAN_g/n64s1/b1/1/gamma:0   (64,)
[TL]   got  14: SRGAN_g/n64s1/c2/1/kernel:0   (3, 3, 64, 64)
[TL]   got  15: SRGAN_g/n64s1/c2/1/bias:0   (64,)
[TL]   got  16: SRGAN_g/n64s1/b2/1/beta:0   (64,)
[TL]   got  17: SRGAN_g/n64s1/b2/1/gamma:0   (64,)
[TL]   got  18: SRGAN_g/n64s1/c1/2/kernel:0   (3, 3, 64, 64)
[TL]   got  19: SRGAN_g/n64s1/c1/2/bias:0   (64,)
[TL]   got  20: SRGAN_g/n64s1/b1/2/beta:0   (64,)
[TL]   got  21: SRGAN_g/n64s1/b1/2/gamma:0   (64,)
[TL]   got  22: SRGAN_g/n64s1/c2/2/kernel:0   (3, 3, 64, 64)
[TL]   got  23: SRGAN_g/n64s1/c2/2/bias:0   (64,)
[TL]   got  24: SRGAN_g/n64s1/b2/2/beta:0   (64,)
[TL]   got  25: SRGAN_g/n64s1/b2/2/gamma:0   (64,)
[TL]   got  26: SRGAN_g/n64s1/c1/3/kernel:0   (3, 3, 64, 64)
[TL]   got  27: SRGAN_g/n64s1/c1/3/bias:0   (64,)
[TL]   got  28: SRGAN_g/n64s1/b1/3/beta:0   (64,)
[TL]   got  29: SRGAN_g/n64s1/b1/3/gamma:0   (64,)
[TL]   got  30: SRGAN_g/n64s1/c2/3/kernel:0   (3, 3, 64, 64)
[TL]   got  31: SRGAN_g/n64s1/c2/3/bias:0   (64,)
[TL]   got  32: SRGAN_g/n64s1/b2/3/beta:0   (64,)
[TL]   got  33: SRGAN_g/n64s1/b2/3/gamma:0   (64,)
[TL]   got  34: SRGAN_g/n64s1/c1/4/kernel:0   (3, 3, 64, 64)
[TL]   got  35: SRGAN_g/n64s1/c1/4/bias:0   (64,)
```

```
[TL]   got  36: SRGAN_g/n64s1/b1/4/beta:0   (64,)
[TL]   got  37: SRGAN_g/n64s1/b1/4/gamma:0   (64,)
[TL]   got  38: SRGAN_g/n64s1/c2/4/kernel:0   (3, 3, 64, 64)
[TL]   got  39: SRGAN_g/n64s1/c2/4/bias:0   (64,)
[TL]   got  40: SRGAN_g/n64s1/b2/4/beta:0   (64,)
[TL]   got  41: SRGAN_g/n64s1/b2/4/gamma:0   (64,)
[TL]   got  42: SRGAN_g/n64s1/c1/5/kernel:0   (3, 3, 64, 64)
[TL]   got  43: SRGAN_g/n64s1/c1/5/bias:0   (64,)
[TL]   got  44: SRGAN_g/n64s1/b1/5/beta:0   (64,)
[TL]   got  45: SRGAN_g/n64s1/b1/5/gamma:0   (64,)
[TL]   got  46: SRGAN_g/n64s1/c2/5/kernel:0   (3, 3, 64, 64)
[TL]   got  47: SRGAN_g/n64s1/c2/5/bias:0   (64,)
[TL]   got  48: SRGAN_g/n64s1/b2/5/beta:0   (64,)
[TL]   got  49: SRGAN_g/n64s1/b2/5/gamma:0   (64,)
[TL]   got  50: SRGAN_g/n64s1/c1/6/kernel:0   (3, 3, 64, 64)
[TL]   got  51: SRGAN_g/n64s1/c1/6/bias:0   (64,)
[TL]   got  52: SRGAN_g/n64s1/b1/6/beta:0   (64,)
[TL]   got  53: SRGAN_g/n64s1/b1/6/gamma:0   (64,)
[TL]   got  54: SRGAN_g/n64s1/c2/6/kernel:0   (3, 3, 64, 64)
[TL]   got  55: SRGAN_g/n64s1/c2/6/bias:0   (64,)
[TL]   got  56: SRGAN_g/n64s1/b2/6/beta:0   (64,)
[TL]   got  57: SRGAN_g/n64s1/b2/6/gamma:0   (64,)
[TL]   got  58: SRGAN_g/n64s1/c1/7/kernel:0   (3, 3, 64, 64)
[TL]   got  59: SRGAN_g/n64s1/c1/7/bias:0   (64,)
[TL]   got  60: SRGAN_g/n64s1/b1/7/beta:0   (64,)
[TL]   got  61: SRGAN_g/n64s1/b1/7/gamma:0   (64,)
[TL]   got  62: SRGAN_g/n64s1/c2/7/kernel:0   (3, 3, 64, 64)
[TL]   got  63: SRGAN_g/n64s1/c2/7/bias:0   (64,)
[TL]   got  64: SRGAN_g/n64s1/b2/7/beta:0   (64,)
[TL]   got  65: SRGAN_g/n64s1/b2/7/gamma:0   (64,)
[TL]   got  66: SRGAN_g/n64s1/c1/8/kernel:0   (3, 3, 64, 64)
[TL]   got  67: SRGAN_g/n64s1/c1/8/bias:0   (64,)
[TL]   got  68: SRGAN_g/n64s1/b1/8/beta:0   (64,)
[TL]   got  69: SRGAN_g/n64s1/b1/8/gamma:0   (64,)
[TL]   got  70: SRGAN_g/n64s1/c2/8/kernel:0   (3, 3, 64, 64)
[TL]   got  71: SRGAN_g/n64s1/c2/8/bias:0   (64,)
[TL]   got  72: SRGAN_g/n64s1/b2/8/beta:0   (64,)
[TL]   got  73: SRGAN_g/n64s1/b2/8/gamma:0   (64,)
[TL]   got  74: SRGAN_g/n64s1/c1/9/kernel:0   (3, 3, 64, 64)
[TL]   got  75: SRGAN_g/n64s1/c1/9/bias:0   (64,)
[TL]   got  76: SRGAN_g/n64s1/b1/9/beta:0   (64,)
[TL]   got  77: SRGAN_g/n64s1/b1/9/gamma:0   (64,)
[TL]   got  78: SRGAN_g/n64s1/c2/9/kernel:0   (3, 3, 64, 64)
[TL]   got  79: SRGAN_g/n64s1/c2/9/bias:0   (64,)
[TL]   got  80: SRGAN_g/n64s1/b2/9/beta:0   (64,)
[TL]   got  81: SRGAN_g/n64s1/b2/9/gamma:0   (64,)
[TL]   got  82: SRGAN_g/n64s1/c1/10/kernel:0   (3, 3, 64, 64)
[TL]   got  83: SRGAN_g/n64s1/c1/10/bias:0   (64,)
[TL]   got  84: SRGAN_g/n64s1/b1/10/beta:0   (64,)
[TL]   got  85: SRGAN_g/n64s1/b1/10/gamma:0   (64,)
[TL]   got  86: SRGAN_g/n64s1/c2/10/kernel:0   (3, 3, 64, 64)
[TL]   got  87: SRGAN_g/n64s1/c2/10/bias:0   (64,)
[TL]   got  88: SRGAN_g/n64s1/b2/10/beta:0   (64,)
[TL]   got  89: SRGAN_g/n64s1/b2/10/gamma:0   (64,)
[TL]   got  90: SRGAN_g/n64s1/c1/11/kernel:0   (3, 3, 64, 64)
[TL]   got  91: SRGAN_g/n64s1/c1/11/bias:0   (64,)
[TL]   got  92: SRGAN_g/n64s1/b1/11/beta:0   (64,)
```

```
[TL]   got  93: SRGAN_g/n64s1/b1/11/gamma:0   (64,)
[TL]   got  94: SRGAN_g/n64s1/c2/11/kernel:0   (3, 3, 64, 64)
[TL]   got  95: SRGAN_g/n64s1/c2/11/bias:0   (64,)
[TL]   got  96: SRGAN_g/n64s1/b2/11/beta:0   (64,)
[TL]   got  97: SRGAN_g/n64s1/b2/11/gamma:0   (64,)
[TL]   got  98: SRGAN_g/n64s1/c1/12/kernel:0   (3, 3, 64, 64)
[TL]   got  99: SRGAN_g/n64s1/c1/12/bias:0   (64,)
[TL]   got 100: SRGAN_g/n64s1/b1/12/beta:0   (64,)
[TL]   got 101: SRGAN_g/n64s1/b1/12/gamma:0   (64,)
[TL]   got 102: SRGAN_g/n64s1/c2/12/kernel:0   (3, 3, 64, 64)
[TL]   got 103: SRGAN_g/n64s1/c2/12/bias:0   (64,)
[TL]   got 104: SRGAN_g/n64s1/b2/12/beta:0   (64,)
[TL]   got 105: SRGAN_g/n64s1/b2/12/gamma:0   (64,)
[TL]   got 106: SRGAN_g/n64s1/c1/13/kernel:0   (3, 3, 64, 64)
[TL]   got 107: SRGAN_g/n64s1/c1/13/bias:0   (64,)
[TL]   got 108: SRGAN_g/n64s1/b1/13/beta:0   (64,)
[TL]   got 109: SRGAN_g/n64s1/b1/13/gamma:0   (64,)
[TL]   got 110: SRGAN_g/n64s1/c2/13/kernel:0   (3, 3, 64, 64)
[TL]   got 111: SRGAN_g/n64s1/c2/13/bias:0   (64,)
[TL]   got 112: SRGAN_g/n64s1/b2/13/beta:0   (64,)
[TL]   got 113: SRGAN_g/n64s1/b2/13/gamma:0   (64,)
[TL]   got 114: SRGAN_g/n64s1/c1/14/kernel:0   (3, 3, 64, 64)
[TL]   got 115: SRGAN_g/n64s1/c1/14/bias:0   (64,)
[TL]   got 116: SRGAN_g/n64s1/b1/14/beta:0   (64,)
[TL]   got 117: SRGAN_g/n64s1/b1/14/gamma:0   (64,)
[TL]   got 118: SRGAN_g/n64s1/c2/14/kernel:0   (3, 3, 64, 64)
[TL]   got 119: SRGAN_g/n64s1/c2/14/bias:0   (64,)
[TL]   got 120: SRGAN_g/n64s1/b2/14/beta:0   (64,)
[TL]   got 121: SRGAN_g/n64s1/b2/14/gamma:0   (64,)
[TL]   got 122: SRGAN_g/n64s1/c1/15/kernel:0   (3, 3, 64, 64)
[TL]   got 123: SRGAN_g/n64s1/c1/15/bias:0   (64,)
[TL]   got 124: SRGAN_g/n64s1/b1/15/beta:0   (64,)
[TL]   got 125: SRGAN_g/n64s1/b1/15/gamma:0   (64,)
[TL]   got 126: SRGAN_g/n64s1/c2/15/kernel:0   (3, 3, 64, 64)
[TL]   got 127: SRGAN_g/n64s1/c2/15/bias:0   (64,)
[TL]   got 128: SRGAN_g/n64s1/b2/15/beta:0   (64,)
[TL]   got 129: SRGAN_g/n64s1/b2/15/gamma:0   (64,)
[TL]   got 130: SRGAN_g/n64s1/c/m/kernel:0   (3, 3, 64, 64)
[TL]   got 131: SRGAN_g/n64s1/c/m/bias:0   (64,)
[TL]   got 132: SRGAN_g/n64s1/b/m/beta:0   (64,)
[TL]   got 133: SRGAN_g/n64s1/b/m/gamma:0   (64,)
[TL]   got 134: SRGAN_g/n256s1/1/kernel:0   (3, 3, 64, 256)
[TL]   got 135: SRGAN_g/n256s1/1/bias:0   (256,)
[TL]   got 136: SRGAN_g/out/kernel:0   (1, 1, 64, 3)
[TL]   got 137: SRGAN_g/out/bias:0   (3,)
[TL]   [*] geting variables with SRGAN_d
[TL]   got   0: SRGAN_d/h0/c/kernel:0   (4, 4, 3, 64)
[TL]   got   1: SRGAN_d/h0/c/bias:0   (64,)
[TL]   got   2: SRGAN_d/h1/c/kernel:0   (4, 4, 64, 128)
[TL]   got   3: SRGAN_d/h1/c/bias:0   (128,)
[TL]   got   4: SRGAN_d/h1/bn/beta:0   (128,)
[TL]   got   5: SRGAN_d/h1/bn/gamma:0   (128,)
[TL]   got   6: SRGAN_d/h2/c/kernel:0   (4, 4, 128, 256)
[TL]   got   7: SRGAN_d/h2/c/bias:0   (256,)
[TL]   got   8: SRGAN_d/h2/bn/beta:0   (256,)
[TL]   got   9: SRGAN_d/h2/bn/gamma:0   (256,)
[TL]   got  10: SRGAN_d/h3/c/kernel:0   (4, 4, 256, 512)
```

```
[TL]   got  11: SRGAN_d/h3/c/bias:0   (512,)
[TL]   got  12: SRGAN_d/h3/bn/beta:0   (512,)
[TL]   got  13: SRGAN_d/h3/bn/gamma:0   (512,)
[TL]   got  14: SRGAN_d/h4/c/kernel:0   (4, 4, 512, 1024)
[TL]   got  15: SRGAN_d/h4/c/bias:0   (1024,)
[TL]   got  16: SRGAN_d/h4/bn/beta:0   (1024,)
[TL]   got  17: SRGAN_d/h4/bn/gamma:0   (1024,)
[TL]   got  18: SRGAN_d/h5/c/kernel:0   (4, 4, 1024, 2048)
[TL]   got  19: SRGAN_d/h5/c/bias:0   (2048,)
[TL]   got  20: SRGAN_d/h5/bn/beta:0   (2048,)
[TL]   got  21: SRGAN_d/h5/bn/gamma:0   (2048,)
[TL]   got  22: SRGAN_d/h6/c/kernel:0   (1, 1, 2048, 1024)
[TL]   got  23: SRGAN_d/h6/c/bias:0   (1024,)
[TL]   got  24: SRGAN_d/h6/bn/beta:0   (1024,)
[TL]   got  25: SRGAN_d/h6/bn/gamma:0   (1024,)
[TL]   got  26: SRGAN_d/h7/c/kernel:0   (1, 1, 1024, 512)
[TL]   got  27: SRGAN_d/h7/c/bias:0   (512,)
[TL]   got  28: SRGAN_d/h7/bn/beta:0   (512,)
[TL]   got  29: SRGAN_d/h7/bn/gamma:0   (512,)
[TL]   got  30: SRGAN_d/res/c/kernel:0   (1, 1, 512, 128)
[TL]   got  31: SRGAN_d/res/c/bias:0   (128,)
[TL]   got  32: SRGAN_d/res/bn/beta:0   (128,)
[TL]   got  33: SRGAN_d/res/bn/gamma:0   (128,)
[TL]   got  34: SRGAN_d/res/c2/kernel:0   (3, 3, 128, 128)
[TL]   got  35: SRGAN_d/res/c2/bias:0   (128,)
[TL]   got  36: SRGAN_d/res/bn2/beta:0   (128,)
[TL]   got  37: SRGAN_d/res/bn2/gamma:0   (128,)
[TL]   got  38: SRGAN_d/res/c3/kernel:0   (3, 3, 128, 512)
[TL]   got  39: SRGAN_d/res/c3/bias:0   (512,)
[TL]   got  40: SRGAN_d/res/bn3/beta:0   (512,)
[TL]   got  41: SRGAN_d/res/bn3/gamma:0   (512,)
[TL]   got  42: SRGAN_d/ho/dense/W:0   (4608, 1)
[TL]   got  43: SRGAN_d/ho/dense/b:0   (1,)
[TL] [*] Load ../output/SRGAN2/checkpoint/g_srgan_init.npz SUCCESS!
[TL] ERROR: file ../output/SRGAN2/checkpoint/d_srgan.npz doesn't exist.
  Loading conv1_1: (3, 3, 3, 64), (64,)
  Loading conv1_2: (3, 3, 64, 64), (64,)
  Loading conv2_1: (3, 3, 64, 128), (128,)
  Loading conv2_2: (3, 3, 128, 128), (128,)
  Loading conv3_1: (3, 3, 128, 256), (256,)
  Loading conv3_2: (3, 3, 256, 256), (256,)
  Loading conv3_3: (3, 3, 256, 256), (256,)
  Loading conv3_4: (3, 3, 256, 256), (256,)
  Loading conv4_1: (3, 3, 256, 512), (512,)
  Loading conv4_2: (3, 3, 512, 512), (512,)
  Loading conv4_3: (3, 3, 512, 512), (512,)
  Loading conv4_4: (3, 3, 512, 512), (512,)
  Loading conv5_1: (3, 3, 512, 512), (512,)
  Loading conv5_2: (3, 3, 512, 512), (512,)
  Loading conv5_3: (3, 3, 512, 512), (512,)
  Loading conv5_4: (3, 3, 512, 512), (512,)
  Loading fc6: (25088, 4096), (4096,)
  Loading fc7: (4096, 4096), (4096,)
  Loading fc8: (4096, 1000), (1000,)
sample LR sub-image: (16, 96, 96, 3) -1.0 1.0
sample HR sub-image: (16, 192, 192, 3) -1.0 1.0
finish saving sample images
```

```
 ** fixed learning rate: 0.000100 (for init G)
Epoch [ 0/ 0]    0 time: 33.24s, mse: 0.0123, psnr: 26.4093
Epoch [ 0/ 0]    1 time: 29.89s, mse: 0.0171, psnr: 24.5435
Epoch [ 0/ 0]    2 time: 30.25s, mse: 0.0188, psnr: 23.8611
Epoch [ 0/ 0]    3 time: 29.72s, mse: 0.0210, psnr: 23.0730
Epoch [ 0/ 0]    4 time: 30.03s, mse: 0.0139, psnr: 24.9418
Epoch [ 0/ 0]    5 time: 30.18s, mse: 0.0175, psnr: 24.2590
Epoch [ 0/ 0]    6 time: 29.92s, mse: 0.0128, psnr: 25.5272
Epoch [ 0/ 0]    7 time: 34.47s, mse: 0.0131, psnr: 25.4647
Epoch [ 0/ 0]    8 time: 33.66s, mse: 0.0139, psnr: 25.3225
Epoch [ 0/ 0]    9 time: 32.54s, mse: 0.0164, psnr: 24.5303
Epoch [ 0/ 0]   10 time: 31.16s, mse: 0.0218, psnr: 24.4800
Epoch [ 0/ 0]   11 time: 31.86s, mse: 0.0188, psnr: 24.2042
Epoch [ 0/ 0]   12 time: 32.25s, mse: 0.0165, psnr: 24.2907
Epoch [ 0/ 0]   13 time: 30.13s, mse: 0.0125, psnr: 25.3944
Epoch [ 0/ 0]   14 time: 29.69s, mse: 0.0128, psnr: 25.3797
Epoch [ 0/ 0]   15 time: 29.70s, mse: 0.0151, psnr: 25.0383
Epoch [ 0/ 0]   16 time: 29.76s, mse: 0.0141, psnr: 25.4026
Epoch [ 0/ 0]   17 time: 31.85s, mse: 0.0127, psnr: 25.7208
Epoch [ 0/ 0]   18 time: 31.46s, mse: 0.0142, psnr: 24.9992
Epoch [ 0/ 0]   19 time: 30.04s, mse: 0.0208, psnr: 23.4861
Epoch [ 0/ 0]   20 time: 29.84s, mse: 0.0171, psnr: 24.1585
Epoch [ 0/ 0]   21 time: 30.02s, mse: 0.0182, psnr: 23.7389
Epoch [ 0/ 0]   22 time: 33.40s, mse: 0.0152, psnr: 25.1947
Epoch [ 0/ 0]   23 time: 35.95s, mse: 0.0133, psnr: 25.6332
Epoch [ 0/ 0]   24 time: 31.86s, mse: 0.0131, psnr: 25.1897
Epoch [ 0/ 0]   25 time: 30.78s, mse: 0.0136, psnr: 25.4094
Epoch [ 0/ 0]   26 time: 32.32s, mse: 0.0230, psnr: 24.1464
Epoch [ 0/ 0]   27 time: 30.66s, mse: 0.0170, psnr: 24.4136
Epoch [ 0/ 0]   28 time: 30.54s, mse: 0.0147, psnr: 24.7636
Epoch [ 0/ 0]   29 time: 31.60s, mse: 0.0171, psnr: 24.0634
Epoch [ 0/ 0]   30 time: 31.50s, mse: 0.0197, psnr: 24.0961
Epoch [ 0/ 0]   31 time: 33.11s, mse: 0.0227, psnr: 22.8146
Epoch [ 0/ 0]   32 time: 32.15s, mse: 0.0146, psnr: 24.7168
Epoch [ 0/ 0]   33 time: 30.46s, mse: 0.0126, psnr: 26.2573
Epoch [ 0/ 0]   34 time: 31.24s, mse: 0.0122, psnr: 26.2371
Epoch [ 0/ 0]   35 time: 30.65s, mse: 0.0177, psnr: 24.2307
Epoch [ 0/ 0]   36 time: 31.43s, mse: 0.0122, psnr: 26.0283
Epoch [ 0/ 0]   37 time: 30.91s, mse: 0.0107, psnr: 26.4816
Epoch [ 0/ 0]   38 time: 31.08s, mse: 0.0138, psnr: 25.3411
Epoch [ 0/ 0]   39 time: 29.68s, mse: 0.0164, psnr: 24.8477
Epoch [ 0/ 0]   40 time: 29.67s, mse: 0.0116, psnr: 25.9916
Epoch [ 0/ 0]   41 time: 31.22s, mse: 0.0129, psnr: 26.1476
Epoch [ 0/ 0]   42 time: 31.31s, mse: 0.0138, psnr: 25.7726
Epoch [ 0/ 0]   43 time: 30.68s, mse: 0.0124, psnr: 25.8166
Epoch [ 0/ 0]   44 time: 30.23s, mse: 0.0126, psnr: 25.8262
Epoch [ 0/ 0]   45 time: 30.99s, mse: 0.0235, psnr: 23.0315
Epoch [ 0/ 0]   46 time: 27.60s, mse: 0.0103, psnr: 26.2994
[*] Epoch: [ 0/ 0] time: 1462.70s, mse: 0.0155, psnr: 24.9569
 ** init lr: 0.000100  decay_every_init: 10, lr_decay: 0.100000 (for GA
N)
Epoch [ 0/20]    0 time: 79.29s, d_loss: 1.5770 g_loss: 0.0572 (mse: 0.
0176, psnr: 25.5059, accuracy: 0.4688)
Epoch [ 0/20]    1 time: 68.73s, d_loss: 2.7956 g_loss: 0.0604 (mse: 0.
0172, psnr: 24.2452, accuracy: 0.5312)
Epoch [ 0/20]    2 time: 68.53s, d_loss: 2.0243 g_loss: 0.0583 (mse: 0.
0190, psnr: 23.4915, accuracy: 0.4531)
```

```
Epoch [ 0/20]    3 time: 67.98s, d_loss: 2.0739 g_loss: 0.0466 (mse: 0.
0146, psnr: 25.0962, accuracy: 0.5469)
Epoch [ 0/20]    4 time: 68.46s, d_loss: 1.9308 g_loss: 0.0447 (mse: 0.
0114, psnr: 26.2912, accuracy: 0.5156)
Epoch [ 0/20]    5 time: 69.32s, d_loss: 2.1796 g_loss: 0.0466 (mse: 0.
0132, psnr: 25.7165, accuracy: 0.4844)
Epoch [ 0/20]    6 time: 66.94s, d_loss: 2.5313 g_loss: 0.0490 (mse: 0.
0156, psnr: 25.0556, accuracy: 0.5312)
Epoch [ 0/20]    7 time: 65.67s, d_loss: 1.8957 g_loss: 0.0446 (mse: 0.
0126, psnr: 25.6632, accuracy: 0.4844)
Epoch [ 0/20]    8 time: 65.46s, d_loss: 1.8393 g_loss: 0.0423 (mse: 0.
0116, psnr: 26.0124, accuracy: 0.5000)
Epoch [ 0/20]    9 time: 65.44s, d_loss: 2.0583 g_loss: 0.0457 (mse: 0.
0135, psnr: 25.5165, accuracy: 0.4844)
Epoch [ 0/20]   10 time: 65.43s, d_loss: 1.8794 g_loss: 0.0378 (mse: 0.
0115, psnr: 25.8179, accuracy: 0.5156)
Epoch [ 0/20]   11 time: 65.74s, d_loss: 1.9780 g_loss: 0.0353 (mse: 0.
0098, psnr: 26.4602, accuracy: 0.5312)
Epoch [ 0/20]   12 time: 65.54s, d_loss: 1.9587 g_loss: 0.0385 (mse: 0.
0118, psnr: 26.3272, accuracy: 0.4531)
Epoch [ 0/20]   13 time: 65.62s, d_loss: 1.8673 g_loss: 0.0332 (mse: 0.
0096, psnr: 27.1684, accuracy: 0.4688)
Epoch [ 0/20]   14 time: 66.11s, d_loss: 1.6567 g_loss: 0.0400 (mse: 0.
0131, psnr: 25.7048, accuracy: 0.5781)
Epoch [ 0/20]   15 time: 65.99s, d_loss: 1.9417 g_loss: 0.0455 (mse: 0.
0183, psnr: 24.5470, accuracy: 0.5312)
Epoch [ 0/20]   16 time: 65.82s, d_loss: 1.9787 g_loss: 0.0504 (mse: 0.
0199, psnr: 24.5038, accuracy: 0.5156)
Epoch [ 0/20]   17 time: 66.08s, d_loss: 1.8170 g_loss: 0.0511 (mse: 0.
0208, psnr: 23.0814, accuracy: 0.4531)
Epoch [ 0/20]   18 time: 66.92s, d_loss: 1.7662 g_loss: 0.0374 (mse: 0.
0114, psnr: 26.5362, accuracy: 0.4844)
Epoch [ 0/20]   19 time: 66.71s, d_loss: 1.8398 g_loss: 0.0401 (mse: 0.
0138, psnr: 25.4356, accuracy: 0.5156)
Epoch [ 0/20]   20 time: 66.80s, d_loss: 1.8167 g_loss: 0.0330 (mse: 0.
0102, psnr: 26.7629, accuracy: 0.5000)
Epoch [ 0/20]   21 time: 65.06s, d_loss: 1.7215 g_loss: 0.0387 (mse: 0.
0115, psnr: 26.3293, accuracy: 0.5000)
Epoch [ 0/20]   22 time: 69.43s, d_loss: 1.6397 g_loss: 0.0430 (mse: 0.
0157, psnr: 24.7524, accuracy: 0.5625)
Epoch [ 0/20]   23 time: 67.99s, d_loss: 1.8901 g_loss: 0.0411 (mse: 0.
0135, psnr: 25.1460, accuracy: 0.4844)
Epoch [ 0/20]   24 time: 65.13s, d_loss: 1.8525 g_loss: 0.0398 (mse: 0.
0129, psnr: 25.5928, accuracy: 0.4844)
Epoch [ 0/20]   25 time: 65.91s, d_loss: 1.7183 g_loss: 0.0396 (mse: 0.
0125, psnr: 26.1091, accuracy: 0.5000)
Epoch [ 0/20]   26 time: 64.99s, d_loss: 1.7965 g_loss: 0.0426 (mse: 0.
0147, psnr: 25.4196, accuracy: 0.5312)
Epoch [ 0/20]   27 time: 66.25s, d_loss: 1.7567 g_loss: 0.0408 (mse: 0.
0155, psnr: 25.4852, accuracy: 0.4844)
Epoch [ 0/20]   28 time: 65.77s, d_loss: 1.6823 g_loss: 0.0282 (mse: 0.
0085, psnr: 27.5555, accuracy: 0.4844)
Epoch [ 0/20]   29 time: 66.61s, d_loss: 1.7713 g_loss: 0.0345 (mse: 0.
0109, psnr: 26.6426, accuracy: 0.5156)
Epoch [ 0/20]   30 time: 66.58s, d_loss: 1.6333 g_loss: 0.0396 (mse: 0.
0166, psnr: 26.3186, accuracy: 0.5156)
Epoch [ 0/20]   31 time: 65.17s, d_loss: 1.7113 g_loss: 0.0285 (mse: 0.
```

```
0095, psnr: 27.2264, accuracy: 0.4531)
Epoch [ 0/20]   32 time: 65.37s, d_loss: 1.6569 g_loss: 0.0389 (mse: 0.
0124, psnr: 25.9464, accuracy: 0.4688)
Epoch [ 0/20]   33 time: 65.25s, d_loss: 1.5607 g_loss: 0.0389 (mse: 0.
0132, psnr: 25.8983, accuracy: 0.5312)
Epoch [ 0/20]   34 time: 66.68s, d_loss: 1.5328 g_loss: 0.0363 (mse: 0.
0119, psnr: 26.0318, accuracy: 0.5000)
Epoch [ 0/20]   35 time: 67.64s, d_loss: 1.6199 g_loss: 0.0365 (mse: 0.
0102, psnr: 26.7410, accuracy: 0.5312)
Epoch [ 0/20]   36 time: 65.79s, d_loss: 1.6835 g_loss: 0.0365 (mse: 0.
0110, psnr: 26.2807, accuracy: 0.5000)
Epoch [ 0/20]   37 time: 65.33s, d_loss: 1.6075 g_loss: 0.0419 (mse: 0.
0157, psnr: 24.7933, accuracy: 0.4844)
Epoch [ 0/20]   38 time: 64.99s, d_loss: 1.6089 g_loss: 0.0360 (mse: 0.
0115, psnr: 26.2222, accuracy: 0.4844)
Epoch [ 0/20]   39 time: 66.08s, d_loss: 1.6734 g_loss: 0.0386 (mse: 0.
0156, psnr: 24.7601, accuracy: 0.5312)
Epoch [ 0/20]   40 time: 65.77s, d_loss: 1.7040 g_loss: 0.0364 (mse: 0.
0127, psnr: 26.1969, accuracy: 0.5000)
Epoch [ 0/20]   41 time: 65.70s, d_loss: 1.5940 g_loss: 0.0373 (mse: 0.
0132, psnr: 25.5111, accuracy: 0.5469)
Epoch [ 0/20]   42 time: 65.18s, d_loss: 1.5282 g_loss: 0.0356 (mse: 0.
0119, psnr: 26.0617, accuracy: 0.5000)
Epoch [ 0/20]   43 time: 65.04s, d_loss: 1.5081 g_loss: 0.0326 (mse: 0.
0116, psnr: 26.0915, accuracy: 0.4844)
Epoch [ 0/20]   44 time: 65.98s, d_loss: 1.5668 g_loss: 0.0352 (mse: 0.
0104, psnr: 26.4622, accuracy: 0.5000)
Epoch [ 0/20]   45 time: 65.47s, d_loss: 1.5520 g_loss: 0.0361 (mse: 0.
0107, psnr: 27.1352, accuracy: 0.5156)
Epoch [ 0/20]   46 time: 56.89s, d_loss: 1.5217 g_loss: 0.0315 (mse: 0.
0114, psnr: 26.1367, accuracy: 0.5000)
[*] Epoch: [ 0/20] time: 3118.64s, d_loss: 1.7978 g_loss: 0.0405 (mse:
0.013232, psnr: 25.7827, accuracy: 0.5030)
Epoch [ 1/20]    0 time: 65.73s, d_loss: 1.6481 g_loss: 0.0406 (mse: 0.
0168, psnr: 25.5941, accuracy: 0.4844)
Epoch [ 1/20]    1 time: 64.95s, d_loss: 1.5532 g_loss: 0.0300 (mse: 0.
0094, psnr: 27.5145, accuracy: 0.4688)
Epoch [ 1/20]    2 time: 65.04s, d_loss: 1.5526 g_loss: 0.0370 (mse: 0.
0137, psnr: 25.6062, accuracy: 0.5156)
Epoch [ 1/20]    3 time: 65.64s, d_loss: 1.5293 g_loss: 0.0368 (mse: 0.
0140, psnr: 26.9652, accuracy: 0.4844)
Epoch [ 1/20]    4 time: 65.87s, d_loss: 1.5664 g_loss: 0.0310 (mse: 0.
0092, psnr: 27.2553, accuracy: 0.5000)
Epoch [ 1/20]    5 time: 64.91s, d_loss: 1.5927 g_loss: 0.0392 (mse: 0.
0140, psnr: 25.2486, accuracy: 0.5156)
Epoch [ 1/20]    6 time: 66.92s, d_loss: 1.5849 g_loss: 0.0336 (mse: 0.
0120, psnr: 25.9948, accuracy: 0.5000)
Epoch [ 1/20]    7 time: 64.43s, d_loss: 1.6410 g_loss: 0.0270 (mse: 0.
0090, psnr: 27.1829, accuracy: 0.5000)
Epoch [ 1/20]    8 time: 64.96s, d_loss: 1.5225 g_loss: 0.0410 (mse: 0.
0142, psnr: 25.2409, accuracy: 0.4844)
Epoch [ 1/20]    9 time: 64.66s, d_loss: 1.5003 g_loss: 0.0356 (mse: 0.
0117, psnr: 26.4836, accuracy: 0.5156)
Epoch [ 1/20]   10 time: 64.94s, d_loss: 1.4898 g_loss: 0.0359 (mse: 0.
0120, psnr: 26.2319, accuracy: 0.5000)
Epoch [ 1/20]   11 time: 65.10s, d_loss: 1.5092 g_loss: 0.0356 (mse: 0.
0124, psnr: 26.2024, accuracy: 0.4688)
```

```
Epoch [ 1/20]   12 time: 65.03s, d_loss: 1.5541 g_loss: 0.0302 (mse: 0.
0101, psnr: 26.5483, accuracy: 0.4844)
Epoch [ 1/20]   13 time: 65.32s, d_loss: 1.5642 g_loss: 0.0397 (mse: 0.
0150, psnr: 25.6931, accuracy: 0.5000)
Epoch [ 1/20]   14 time: 64.80s, d_loss: 1.5551 g_loss: 0.0360 (mse: 0.
0133, psnr: 25.5280, accuracy: 0.5469)
Epoch [ 1/20]   15 time: 64.86s, d_loss: 1.5544 g_loss: 0.0339 (mse: 0.
0135, psnr: 25.8344, accuracy: 0.5312)
Epoch [ 1/20]   16 time: 64.77s, d_loss: 1.5629 g_loss: 0.0316 (mse: 0.
0123, psnr: 25.8743, accuracy: 0.5312)
Epoch [ 1/20]   17 time: 64.98s, d_loss: 1.5821 g_loss: 0.0363 (mse: 0.
0133, psnr: 25.5762, accuracy: 0.5312)
Epoch [ 1/20]   18 time: 65.05s, d_loss: 1.6522 g_loss: 0.0354 (mse: 0.
0134, psnr: 25.3837, accuracy: 0.5312)
Epoch [ 1/20]   19 time: 64.91s, d_loss: 1.5231 g_loss: 0.0376 (mse: 0.
0131, psnr: 25.3892, accuracy: 0.5469)
Epoch [ 1/20]   20 time: 64.93s, d_loss: 1.4932 g_loss: 0.0410 (mse: 0.
0165, psnr: 24.4863, accuracy: 0.5156)
Epoch [ 1/20]   21 time: 65.11s, d_loss: 1.5373 g_loss: 0.0350 (mse: 0.
0136, psnr: 25.8327, accuracy: 0.4844)
Epoch [ 1/20]   22 time: 64.94s, d_loss: 1.5437 g_loss: 0.0295 (mse: 0.
0096, psnr: 27.0769, accuracy: 0.5000)
Epoch [ 1/20]   23 time: 64.74s, d_loss: 1.5269 g_loss: 0.0359 (mse: 0.
0114, psnr: 26.1636, accuracy: 0.5000)
Epoch [ 1/20]   24 time: 65.15s, d_loss: 1.4310 g_loss: 0.0379 (mse: 0.
0131, psnr: 25.3872, accuracy: 0.5156)
Epoch [ 1/20]   25 time: 65.03s, d_loss: 1.4410 g_loss: 0.0359 (mse: 0.
0133, psnr: 25.9844, accuracy: 0.5156)
Epoch [ 1/20]   26 time: 64.65s, d_loss: 1.5232 g_loss: 0.0350 (mse: 0.
0109, psnr: 26.1338, accuracy: 0.5156)
Epoch [ 1/20]   27 time: 64.60s, d_loss: 1.5742 g_loss: 0.0423 (mse: 0.
0153, psnr: 25.1010, accuracy: 0.5312)
Epoch [ 1/20]   28 time: 64.55s, d_loss: 1.5920 g_loss: 0.0347 (mse: 0.
0117, psnr: 26.6524, accuracy: 0.4844)
Epoch [ 1/20]   29 time: 64.89s, d_loss: 1.4678 g_loss: 0.0366 (mse: 0.
0125, psnr: 25.9330, accuracy: 0.5469)
Epoch [ 1/20]   30 time: 64.82s, d_loss: 1.5702 g_loss: 0.0372 (mse: 0.
0137, psnr: 25.6285, accuracy: 0.5312)
Epoch [ 1/20]   31 time: 64.64s, d_loss: 1.6014 g_loss: 0.0377 (mse: 0.
0148, psnr: 25.8291, accuracy: 0.5000)
Epoch [ 1/20]   32 time: 64.63s, d_loss: 1.5147 g_loss: 0.0330 (mse: 0.
0109, psnr: 26.4672, accuracy: 0.5312)
Epoch [ 1/20]   33 time: 64.78s, d_loss: 1.5593 g_loss: 0.0303 (mse: 0.
0112, psnr: 26.4520, accuracy: 0.5156)
Epoch [ 1/20]   34 time: 64.71s, d_loss: 1.5920 g_loss: 0.0404 (mse: 0.
0137, psnr: 25.6440, accuracy: 0.5156)
Epoch [ 1/20]   35 time: 64.79s, d_loss: 1.5754 g_loss: 0.0389 (mse: 0.
0156, psnr: 24.8518, accuracy: 0.5000)
Epoch [ 1/20]   36 time: 64.75s, d_loss: 1.4938 g_loss: 0.0373 (mse: 0.
0129, psnr: 25.9227, accuracy: 0.4688)
Epoch [ 1/20]   37 time: 65.29s, d_loss: 1.4901 g_loss: 0.0316 (mse: 0.
0102, psnr: 26.4689, accuracy: 0.5000)
Epoch [ 1/20]   38 time: 64.88s, d_loss: 1.5284 g_loss: 0.0273 (mse: 0.
0099, psnr: 26.5657, accuracy: 0.5000)
Epoch [ 1/20]   39 time: 65.22s, d_loss: 1.5924 g_loss: 0.0392 (mse: 0.
0156, psnr: 24.7541, accuracy: 0.5000)
Epoch [ 1/20]   40 time: 65.06s, d_loss: 1.4988 g_loss: 0.0330 (mse: 0.
```

```
0107, psnr: 26.4667, accuracy: 0.5156)
Epoch [ 1/20]   41 time: 64.79s, d_loss: 1.5985 g_loss: 0.0271 (mse: 0.
0098, psnr: 27.4986, accuracy: 0.5000)
Epoch [ 1/20]   42 time: 65.05s, d_loss: 1.5360 g_loss: 0.0282 (mse: 0.
0103, psnr: 26.3676, accuracy: 0.5000)
Epoch [ 1/20]   43 time: 65.18s, d_loss: 1.5738 g_loss: 0.0343 (mse: 0.
0125, psnr: 25.8338, accuracy: 0.4688)
Epoch [ 1/20]   44 time: 65.10s, d_loss: 1.4802 g_loss: 0.0425 (mse: 0.
0138, psnr: 25.4019, accuracy: 0.5156)
Epoch [ 1/20]   45 time: 64.64s, d_loss: 1.4977 g_loss: 0.0363 (mse: 0.
0169, psnr: 24.2846, accuracy: 0.4688)
Epoch [ 1/20]   46 time: 56.48s, d_loss: 1.4746 g_loss: 0.0305 (mse: 0.
0110, psnr: 26.6935, accuracy: 0.5000)
[*] Epoch: [ 1/20] time: 3046.28s, d_loss: 1.5435 g_loss: 0.0352 (mse:
 0.012628, psnr: 25.9836, accuracy: 0.5060)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 2/20]    0 time: 66.17s, d_loss: 1.5165 g_loss: 0.0334 (mse: 0.
0116, psnr: 25.9833, accuracy: 0.4844)
Epoch [ 2/20]    1 time: 64.67s, d_loss: 1.4462 g_loss: 0.0339 (mse: 0.
0119, psnr: 26.4323, accuracy: 0.5312)
Epoch [ 2/20]    2 time: 64.33s, d_loss: 1.6231 g_loss: 0.0275 (mse: 0.
0096, psnr: 26.5413, accuracy: 0.4688)
Epoch [ 2/20]    3 time: 64.54s, d_loss: 1.5106 g_loss: 0.0272 (mse: 0.
0092, psnr: 27.0935, accuracy: 0.5156)
Epoch [ 2/20]    4 time: 64.47s, d_loss: 1.5606 g_loss: 0.0342 (mse: 0.
0136, psnr: 25.8078, accuracy: 0.5469)
Epoch [ 2/20]    5 time: 64.40s, d_loss: 1.4969 g_loss: 0.0365 (mse: 0.
0139, psnr: 25.4275, accuracy: 0.5312)
Epoch [ 2/20]    6 time: 64.57s, d_loss: 1.4121 g_loss: 0.0340 (mse: 0.
0130, psnr: 26.6138, accuracy: 0.5156)
Epoch [ 2/20]    7 time: 64.36s, d_loss: 1.4560 g_loss: 0.0281 (mse: 0.
0110, psnr: 26.6121, accuracy: 0.5000)
Epoch [ 2/20]    8 time: 64.65s, d_loss: 1.5105 g_loss: 0.0412 (mse: 0.
0142, psnr: 25.3294, accuracy: 0.5156)
Epoch [ 2/20]    9 time: 65.19s, d_loss: 1.5532 g_loss: 0.0315 (mse: 0.
0125, psnr: 25.8826, accuracy: 0.5156)
Epoch [ 2/20]   10 time: 64.69s, d_loss: 1.5024 g_loss: 0.0326 (mse: 0.
0111, psnr: 26.3962, accuracy: 0.5156)
Epoch [ 2/20]   11 time: 65.08s, d_loss: 1.4074 g_loss: 0.0425 (mse: 0.
0183, psnr: 24.5364, accuracy: 0.5000)
Epoch [ 2/20]   12 time: 64.78s, d_loss: 1.5286 g_loss: 0.0354 (mse: 0.
0135, psnr: 25.4994, accuracy: 0.5000)
Epoch [ 2/20]   13 time: 65.00s, d_loss: 1.5818 g_loss: 0.0317 (mse: 0.
0116, psnr: 26.3913, accuracy: 0.5156)
Epoch [ 2/20]   14 time: 64.91s, d_loss: 1.4699 g_loss: 0.0352 (mse: 0.
0116, psnr: 26.1028, accuracy: 0.4531)
Epoch [ 2/20]   15 time: 64.70s, d_loss: 1.4594 g_loss: 0.0324 (mse: 0.
0113, psnr: 26.6607, accuracy: 0.5000)
Epoch [ 2/20]   16 time: 64.87s, d_loss: 1.4833 g_loss: 0.0329 (mse: 0.
0125, psnr: 25.9949, accuracy: 0.5000)
Epoch [ 2/20]   17 time: 64.82s, d_loss: 1.5210 g_loss: 0.0397 (mse: 0.
0191, psnr: 24.5141, accuracy: 0.5312)
Epoch [ 2/20]   18 time: 65.13s, d_loss: 1.4564 g_loss: 0.0323 (mse: 0.
```

```
0114, psnr: 26.2776, accuracy: 0.5000)
Epoch [ 2/20]   19 time: 64.98s, d_loss: 1.4388 g_loss: 0.0453 (mse: 0.
0248, psnr: 24.1996, accuracy: 0.5156)
Epoch [ 2/20]   20 time: 65.24s, d_loss: 1.4791 g_loss: 0.0347 (mse: 0.
0128, psnr: 25.6780, accuracy: 0.4688)
Epoch [ 2/20]   21 time: 65.43s, d_loss: 1.4709 g_loss: 0.0332 (mse: 0.
0124, psnr: 25.6689, accuracy: 0.4844)
Epoch [ 2/20]   22 time: 64.73s, d_loss: 1.4773 g_loss: 0.0340 (mse: 0.
0117, psnr: 25.9116, accuracy: 0.4844)
Epoch [ 2/20]   23 time: 64.69s, d_loss: 1.5073 g_loss: 0.0348 (mse: 0.
0120, psnr: 26.0425, accuracy: 0.5156)
Epoch [ 2/20]   24 time: 64.71s, d_loss: 1.5167 g_loss: 0.0374 (mse: 0.
0137, psnr: 25.7381, accuracy: 0.5156)
Epoch [ 2/20]   25 time: 64.67s, d_loss: 1.4717 g_loss: 0.0272 (mse: 0.
0096, psnr: 27.6876, accuracy: 0.5000)
Epoch [ 2/20]   26 time: 64.98s, d_loss: 1.4588 g_loss: 0.0377 (mse: 0.
0143, psnr: 24.9999, accuracy: 0.5000)
Epoch [ 2/20]   27 time: 65.09s, d_loss: 1.4581 g_loss: 0.0362 (mse: 0.
0143, psnr: 24.7371, accuracy: 0.5000)
Epoch [ 2/20]   28 time: 64.98s, d_loss: 1.5307 g_loss: 0.0290 (mse: 0.
0101, psnr: 27.2325, accuracy: 0.5156)
Epoch [ 2/20]   29 time: 64.87s, d_loss: 1.4007 g_loss: 0.0407 (mse: 0.
0178, psnr: 24.2114, accuracy: 0.5156)
Epoch [ 2/20]   30 time: 65.06s, d_loss: 1.4255 g_loss: 0.0462 (mse: 0.
0163, psnr: 24.4082, accuracy: 0.5156)
Epoch [ 2/20]   31 time: 64.89s, d_loss: 1.4685 g_loss: 0.0317 (mse: 0.
0114, psnr: 26.2348, accuracy: 0.4844)
Epoch [ 2/20]   32 time: 64.70s, d_loss: 1.4353 g_loss: 0.0361 (mse: 0.
0127, psnr: 25.9996, accuracy: 0.5312)
Epoch [ 2/20]   33 time: 64.82s, d_loss: 1.4362 g_loss: 0.0367 (mse: 0.
0130, psnr: 25.7670, accuracy: 0.4844)
Epoch [ 2/20]   34 time: 65.12s, d_loss: 1.4808 g_loss: 0.0361 (mse: 0.
0137, psnr: 25.6534, accuracy: 0.5000)
Epoch [ 2/20]   35 time: 64.99s, d_loss: 1.4468 g_loss: 0.0313 (mse: 0.
0115, psnr: 26.3659, accuracy: 0.4844)
Epoch [ 2/20]   36 time: 64.91s, d_loss: 1.4737 g_loss: 0.0313 (mse: 0.
0114, psnr: 26.1364, accuracy: 0.4844)
Epoch [ 2/20]   37 time: 64.94s, d_loss: 1.4396 g_loss: 0.0343 (mse: 0.
0140, psnr: 25.6470, accuracy: 0.5000)
Epoch [ 2/20]   38 time: 64.99s, d_loss: 1.4412 g_loss: 0.0475 (mse: 0.
0284, psnr: 25.3422, accuracy: 0.5000)
Epoch [ 2/20]   39 time: 65.34s, d_loss: 1.5099 g_loss: 0.0299 (mse: 0.
0113, psnr: 26.9291, accuracy: 0.5000)
Epoch [ 2/20]   40 time: 64.97s, d_loss: 1.4299 g_loss: 0.0301 (mse: 0.
0115, psnr: 26.3879, accuracy: 0.4844)
Epoch [ 2/20]   41 time: 64.86s, d_loss: 1.4455 g_loss: 0.0294 (mse: 0.
0111, psnr: 26.6019, accuracy: 0.5000)
Epoch [ 2/20]   42 time: 64.90s, d_loss: 1.4832 g_loss: 0.0356 (mse: 0.
0122, psnr: 25.8732, accuracy: 0.5000)
Epoch [ 2/20]   43 time: 65.04s, d_loss: 1.4126 g_loss: 0.0416 (mse: 0.
0158, psnr: 24.8502, accuracy: 0.5312)
Epoch [ 2/20]   44 time: 65.05s, d_loss: 1.4307 g_loss: 0.0388 (mse: 0.
0148, psnr: 25.1455, accuracy: 0.5156)
Epoch [ 2/20]   45 time: 65.01s, d_loss: 1.4371 g_loss: 0.0309 (mse: 0.
0113, psnr: 26.2216, accuracy: 0.5156)
Epoch [ 2/20]   46 time: 56.98s, d_loss: 1.4290 g_loss: 0.0397 (mse: 0.
0144, psnr: 25.3661, accuracy: 0.5536)
```

```
[*] Epoch: [ 2/20] time: 3042.31s, d_loss: 1.4751 g_loss: 0.0349 (mse:
 0.013386, psnr: 25.8539, accuracy: 0.5051)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 3/20]    0 time: 65.54s, d_loss: 1.3976 g_loss: 0.0344 (mse: 0.
0136, psnr: 25.4271, accuracy: 0.5000)
Epoch [ 3/20]    1 time: 64.77s, d_loss: 1.4116 g_loss: 0.0344 (mse: 0.
0132, psnr: 25.9265, accuracy: 0.5000)
Epoch [ 3/20]    2 time: 64.90s, d_loss: 1.4581 g_loss: 0.0349 (mse: 0.
0132, psnr: 25.5559, accuracy: 0.5000)
Epoch [ 3/20]    3 time: 64.97s, d_loss: 1.4198 g_loss: 0.0313 (mse: 0.
0112, psnr: 26.3752, accuracy: 0.5156)
Epoch [ 3/20]    4 time: 64.82s, d_loss: 1.4377 g_loss: 0.0361 (mse: 0.
0154, psnr: 26.2830, accuracy: 0.5000)
Epoch [ 3/20]    5 time: 65.02s, d_loss: 1.4602 g_loss: 0.0408 (mse: 0.
0135, psnr: 25.7266, accuracy: 0.5156)
Epoch [ 3/20]    6 time: 64.98s, d_loss: 1.4598 g_loss: 0.0324 (mse: 0.
0135, psnr: 25.9673, accuracy: 0.5156)
Epoch [ 3/20]    7 time: 65.08s, d_loss: 1.5121 g_loss: 0.0268 (mse: 0.
0097, psnr: 26.8688, accuracy: 0.5156)
Epoch [ 3/20]    8 time: 64.99s, d_loss: 1.4367 g_loss: 0.0352 (mse: 0.
0127, psnr: 25.6898, accuracy: 0.5000)
Epoch [ 3/20]    9 time: 65.32s, d_loss: 1.3962 g_loss: 0.0436 (mse: 0.
0184, psnr: 25.1388, accuracy: 0.5000)
Epoch [ 3/20]   10 time: 64.91s, d_loss: 1.4293 g_loss: 0.0280 (mse: 0.
0104, psnr: 26.9415, accuracy: 0.4844)
Epoch [ 3/20]   11 time: 64.39s, d_loss: 1.4789 g_loss: 0.0431 (mse: 0.
0194, psnr: 25.3444, accuracy: 0.5000)
Epoch [ 3/20]   12 time: 64.63s, d_loss: 1.4631 g_loss: 0.0338 (mse: 0.
0140, psnr: 25.5453, accuracy: 0.5000)
Epoch [ 3/20]   13 time: 64.79s, d_loss: 1.4202 g_loss: 0.0270 (mse: 0.
0101, psnr: 26.4992, accuracy: 0.4844)
Epoch [ 3/20]   14 time: 64.79s, d_loss: 1.4686 g_loss: 0.0373 (mse: 0.
0173, psnr: 23.9923, accuracy: 0.5312)
Epoch [ 3/20]   15 time: 64.86s, d_loss: 1.4181 g_loss: 0.0306 (mse: 0.
0130, psnr: 25.5260, accuracy: 0.4844)
Epoch [ 3/20]   16 time: 64.89s, d_loss: 1.4255 g_loss: 0.0333 (mse: 0.
0115, psnr: 26.3304, accuracy: 0.5312)
Epoch [ 3/20]   17 time: 64.75s, d_loss: 1.4664 g_loss: 0.0404 (mse: 0.
0135, psnr: 25.2608, accuracy: 0.4844)
Epoch [ 3/20]   18 time: 64.50s, d_loss: 1.4651 g_loss: 0.0314 (mse: 0.
0114, psnr: 26.3276, accuracy: 0.5000)
Epoch [ 3/20]   19 time: 65.11s, d_loss: 1.4005 g_loss: 0.0335 (mse: 0.
0111, psnr: 26.5500, accuracy: 0.5156)
Epoch [ 3/20]   20 time: 64.98s, d_loss: 1.4583 g_loss: 0.0352 (mse: 0.
0128, psnr: 26.0199, accuracy: 0.5000)
Epoch [ 3/20]   21 time: 65.00s, d_loss: 1.4809 g_loss: 0.0343 (mse: 0.
0159, psnr: 25.9502, accuracy: 0.5156)
Epoch [ 3/20]   22 time: 64.66s, d_loss: 1.4551 g_loss: 0.0276 (mse: 0.
0096, psnr: 26.8835, accuracy: 0.5156)
Epoch [ 3/20]   23 time: 64.82s, d_loss: 1.4448 g_loss: 0.0366 (mse: 0.
0140, psnr: 24.9305, accuracy: 0.5000)
Epoch [ 3/20]   24 time: 64.88s, d_loss: 1.4033 g_loss: 0.0399 (mse: 0.
0155, psnr: 25.1402, accuracy: 0.5312)
```

```
Epoch [ 3/20]   25 time: 64.91s, d_loss: 1.4647 g_loss: 0.0332 (mse: 0.
0114, psnr: 26.2804, accuracy: 0.5000)
Epoch [ 3/20]   26 time: 64.74s, d_loss: 1.4528 g_loss: 0.0350 (mse: 0.
0128, psnr: 25.7077, accuracy: 0.5000)
Epoch [ 3/20]   27 time: 64.80s, d_loss: 1.4464 g_loss: 0.0334 (mse: 0.
0126, psnr: 26.3745, accuracy: 0.4688)
Epoch [ 3/20]   28 time: 64.90s, d_loss: 1.4526 g_loss: 0.0301 (mse: 0.
0099, psnr: 26.5948, accuracy: 0.5000)
Epoch [ 3/20]   29 time: 64.61s, d_loss: 1.4530 g_loss: 0.0328 (mse: 0.
0124, psnr: 26.9297, accuracy: 0.5000)
Epoch [ 3/20]   30 time: 64.86s, d_loss: 1.4724 g_loss: 0.0327 (mse: 0.
0132, psnr: 26.0512, accuracy: 0.5000)
Epoch [ 3/20]   31 time: 64.96s, d_loss: 1.4494 g_loss: 0.0351 (mse: 0.
0147, psnr: 24.8986, accuracy: 0.5312)
Epoch [ 3/20]   32 time: 64.81s, d_loss: 1.4690 g_loss: 0.0270 (mse: 0.
0096, psnr: 27.0735, accuracy: 0.4844)
Epoch [ 3/20]   33 time: 64.84s, d_loss: 1.4376 g_loss: 0.0305 (mse: 0.
0105, psnr: 26.6018, accuracy: 0.5000)
Epoch [ 3/20]   34 time: 64.64s, d_loss: 1.4369 g_loss: 0.0250 (mse: 0.
0082, psnr: 27.7158, accuracy: 0.4844)
Epoch [ 3/20]   35 time: 64.51s, d_loss: 1.4367 g_loss: 0.0289 (mse: 0.
0138, psnr: 25.6790, accuracy: 0.5156)
Epoch [ 3/20]   36 time: 64.85s, d_loss: 1.4340 g_loss: 0.0341 (mse: 0.
0159, psnr: 26.3964, accuracy: 0.5312)
Epoch [ 3/20]   37 time: 64.86s, d_loss: 1.4422 g_loss: 0.0279 (mse: 0.
0115, psnr: 26.2846, accuracy: 0.5312)
Epoch [ 3/20]   38 time: 64.65s, d_loss: 1.4143 g_loss: 0.0326 (mse: 0.
0118, psnr: 25.9627, accuracy: 0.5000)
Epoch [ 3/20]   39 time: 64.70s, d_loss: 1.4235 g_loss: 0.0324 (mse: 0.
0130, psnr: 25.5178, accuracy: 0.5000)
Epoch [ 3/20]   40 time: 64.84s, d_loss: 1.3890 g_loss: 0.0328 (mse: 0.
0122, psnr: 26.1785, accuracy: 0.5000)
Epoch [ 3/20]   41 time: 64.81s, d_loss: 1.4346 g_loss: 0.0366 (mse: 0.
0145, psnr: 24.9488, accuracy: 0.5000)
Epoch [ 3/20]   42 time: 64.61s, d_loss: 1.5073 g_loss: 0.0257 (mse: 0.
0081, psnr: 27.9699, accuracy: 0.5000)
Epoch [ 3/20]   43 time: 64.81s, d_loss: 1.4336 g_loss: 0.0294 (mse: 0.
0114, psnr: 26.2307, accuracy: 0.5000)
Epoch [ 3/20]   44 time: 64.59s, d_loss: 1.4309 g_loss: 0.0342 (mse: 0.
0123, psnr: 26.1630, accuracy: 0.5000)
Epoch [ 3/20]   45 time: 64.49s, d_loss: 1.4299 g_loss: 0.0310 (mse: 0.
0125, psnr: 25.5269, accuracy: 0.5156)
Epoch [ 3/20]   46 time: 56.60s, d_loss: 1.4630 g_loss: 0.0329 (mse: 0.
0110, psnr: 26.5838, accuracy: 0.5000)
[*] Epoch: [ 3/20] time: 3038.74s, d_loss: 1.4434 g_loss: 0.0331 (mse:
 0.012703, psnr: 26.0398, accuracy: 0.5043)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 4/20]    0 time: 64.95s, d_loss: 1.4629 g_loss: 0.0347 (mse: 0.
0126, psnr: 25.7063, accuracy: 0.4844)
Epoch [ 4/20]    1 time: 64.60s, d_loss: 1.4052 g_loss: 0.0290 (mse: 0.
0103, psnr: 26.8156, accuracy: 0.5312)
Epoch [ 4/20]    2 time: 64.70s, d_loss: 1.4130 g_loss: 0.0382 (mse: 0.
0186, psnr: 24.5536, accuracy: 0.5156)
```

```
Epoch [ 4/20]    3 time: 64.78s, d_loss: 1.4411 g_loss: 0.0322 (mse: 0.
0127, psnr: 25.5927, accuracy: 0.4219)
Epoch [ 4/20]    4 time: 64.83s, d_loss: 1.4597 g_loss: 0.0290 (mse: 0.
0115, psnr: 26.1991, accuracy: 0.5156)
Epoch [ 4/20]    5 time: 64.63s, d_loss: 1.4368 g_loss: 0.0386 (mse: 0.
0142, psnr: 25.7175, accuracy: 0.5156)
Epoch [ 4/20]    6 time: 64.65s, d_loss: 1.4313 g_loss: 0.0329 (mse: 0.
0115, psnr: 26.2755, accuracy: 0.5000)
Epoch [ 4/20]    7 time: 64.89s, d_loss: 1.4580 g_loss: 0.0295 (mse: 0.
0107, psnr: 26.6827, accuracy: 0.5156)
Epoch [ 4/20]    8 time: 64.86s, d_loss: 1.4517 g_loss: 0.0334 (mse: 0.
0142, psnr: 26.4135, accuracy: 0.5000)
Epoch [ 4/20]    9 time: 64.69s, d_loss: 1.4438 g_loss: 0.0306 (mse: 0.
0103, psnr: 26.8911, accuracy: 0.5156)
Epoch [ 4/20]   10 time: 64.78s, d_loss: 1.4631 g_loss: 0.0317 (mse: 0.
0114, psnr: 26.5635, accuracy: 0.5156)
Epoch [ 4/20]   11 time: 64.71s, d_loss: 1.4835 g_loss: 0.0372 (mse: 0.
0157, psnr: 25.0120, accuracy: 0.5000)
Epoch [ 4/20]   12 time: 64.71s, d_loss: 1.3914 g_loss: 0.0398 (mse: 0.
0197, psnr: 24.0049, accuracy: 0.5156)
Epoch [ 4/20]   13 time: 64.68s, d_loss: 1.4859 g_loss: 0.0270 (mse: 0.
0088, psnr: 27.4243, accuracy: 0.5000)
Epoch [ 4/20]   14 time: 65.11s, d_loss: 1.4382 g_loss: 0.0285 (mse: 0.
0098, psnr: 27.2169, accuracy: 0.5000)
Epoch [ 4/20]   15 time: 64.71s, d_loss: 1.4131 g_loss: 0.0255 (mse: 0.
0091, psnr: 26.9637, accuracy: 0.5312)
Epoch [ 4/20]   16 time: 64.60s, d_loss: 1.4140 g_loss: 0.0261 (mse: 0.
0095, psnr: 27.2727, accuracy: 0.4844)
Epoch [ 4/20]   17 time: 65.02s, d_loss: 1.4415 g_loss: 0.0390 (mse: 0.
0198, psnr: 26.4894, accuracy: 0.5000)
Epoch [ 4/20]   18 time: 64.99s, d_loss: 1.4628 g_loss: 0.0337 (mse: 0.
0125, psnr: 26.1668, accuracy: 0.5000)
Epoch [ 4/20]   19 time: 64.85s, d_loss: 1.4522 g_loss: 0.0315 (mse: 0.
0110, psnr: 26.7544, accuracy: 0.5312)
Epoch [ 4/20]   20 time: 65.26s, d_loss: 1.4501 g_loss: 0.0314 (mse: 0.
0111, psnr: 26.7350, accuracy: 0.5625)
Epoch [ 4/20]   21 time: 64.71s, d_loss: 1.4502 g_loss: 0.0279 (mse: 0.
0096, psnr: 27.3264, accuracy: 0.5156)
Epoch [ 4/20]   22 time: 64.71s, d_loss: 1.4061 g_loss: 0.0331 (mse: 0.
0123, psnr: 25.9103, accuracy: 0.5000)
Epoch [ 4/20]   23 time: 64.74s, d_loss: 1.5133 g_loss: 0.0338 (mse: 0.
0132, psnr: 25.4397, accuracy: 0.5000)
Epoch [ 4/20]   24 time: 64.62s, d_loss: 1.4511 g_loss: 0.0275 (mse: 0.
0104, psnr: 26.9748, accuracy: 0.5000)
Epoch [ 4/20]   25 time: 64.73s, d_loss: 1.4332 g_loss: 0.0330 (mse: 0.
0118, psnr: 26.1065, accuracy: 0.4844)
Epoch [ 4/20]   26 time: 64.52s, d_loss: 1.4281 g_loss: 0.0369 (mse: 0.
0126, psnr: 25.8794, accuracy: 0.5000)
Epoch [ 4/20]   27 time: 64.76s, d_loss: 1.4434 g_loss: 0.0276 (mse: 0.
0099, psnr: 27.1938, accuracy: 0.5156)
Epoch [ 4/20]   28 time: 64.59s, d_loss: 1.4373 g_loss: 0.0296 (mse: 0.
0108, psnr: 26.7125, accuracy: 0.5156)
Epoch [ 4/20]   29 time: 64.59s, d_loss: 1.4755 g_loss: 0.0396 (mse: 0.
0140, psnr: 25.1671, accuracy: 0.5000)
Epoch [ 4/20]   30 time: 64.68s, d_loss: 1.4462 g_loss: 0.0299 (mse: 0.
0111, psnr: 26.6776, accuracy: 0.5000)
Epoch [ 4/20]   31 time: 64.72s, d_loss: 1.4446 g_loss: 0.0226 (mse: 0.
```

```
0072, psnr: 28.0880, accuracy: 0.5000)
Epoch [ 4/20]   32 time: 64.64s, d_loss: 1.4150 g_loss: 0.0330 (mse: 0.
0140, psnr: 26.0378, accuracy: 0.5312)
Epoch [ 4/20]   33 time: 64.75s, d_loss: 1.4448 g_loss: 0.0315 (mse: 0.
0121, psnr: 26.7085, accuracy: 0.5000)
Epoch [ 4/20]   34 time: 64.73s, d_loss: 1.4205 g_loss: 0.0368 (mse: 0.
0150, psnr: 25.3832, accuracy: 0.5156)
Epoch [ 4/20]   35 time: 64.68s, d_loss: 1.4200 g_loss: 0.0324 (mse: 0.
0103, psnr: 26.6567, accuracy: 0.5000)
Epoch [ 4/20]   36 time: 64.75s, d_loss: 1.4170 g_loss: 0.0314 (mse: 0.
0124, psnr: 26.4337, accuracy: 0.5156)
Epoch [ 4/20]   37 time: 64.57s, d_loss: 1.4050 g_loss: 0.0287 (mse: 0.
0109, psnr: 26.7812, accuracy: 0.5312)
Epoch [ 4/20]   38 time: 64.54s, d_loss: 1.4669 g_loss: 0.0233 (mse: 0.
0077, psnr: 28.0447, accuracy: 0.4844)
Epoch [ 4/20]   39 time: 64.47s, d_loss: 1.4399 g_loss: 0.0250 (mse: 0.
0083, psnr: 27.6213, accuracy: 0.5156)
Epoch [ 4/20]   40 time: 64.83s, d_loss: 1.4151 g_loss: 0.0322 (mse: 0.
0115, psnr: 26.5260, accuracy: 0.5000)
Epoch [ 4/20]   41 time: 64.61s, d_loss: 1.4146 g_loss: 0.0306 (mse: 0.
0100, psnr: 27.0060, accuracy: 0.5312)
Epoch [ 4/20]   42 time: 64.73s, d_loss: 1.4303 g_loss: 0.0407 (mse: 0.
0195, psnr: 23.5588, accuracy: 0.5625)
Epoch [ 4/20]   43 time: 64.61s, d_loss: 1.4132 g_loss: 0.0318 (mse: 0.
0129, psnr: 25.7609, accuracy: 0.5000)
Epoch [ 4/20]   44 time: 64.45s, d_loss: 1.3994 g_loss: 0.0289 (mse: 0.
0117, psnr: 25.9561, accuracy: 0.5156)
Epoch [ 4/20]   45 time: 64.73s, d_loss: 1.4617 g_loss: 0.0282 (mse: 0.
0091, psnr: 27.3120, accuracy: 0.5000)
Epoch [ 4/20]   46 time: 56.59s, d_loss: 1.4399 g_loss: 0.0279 (mse: 0.
0094, psnr: 27.4121, accuracy: 0.5000)
[*] Epoch: [ 4/20] time: 3034.05s, d_loss: 1.4390 g_loss: 0.0316 (mse:
 0.011972, psnr: 26.3857, accuracy: 0.5083)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 5/20]    0 time: 64.63s, d_loss: 1.4023 g_loss: 0.0260 (mse: 0.
0103, psnr: 26.4914, accuracy: 0.5312)
Epoch [ 5/20]    1 time: 64.63s, d_loss: 1.4240 g_loss: 0.0361 (mse: 0.
0133, psnr: 25.4258, accuracy: 0.4844)
Epoch [ 5/20]    2 time: 64.66s, d_loss: 1.4545 g_loss: 0.0316 (mse: 0.
0117, psnr: 26.2962, accuracy: 0.5156)
Epoch [ 5/20]    3 time: 64.72s, d_loss: 1.3990 g_loss: 0.0235 (mse: 0.
0080, psnr: 27.9191, accuracy: 0.5000)
Epoch [ 5/20]    4 time: 64.65s, d_loss: 1.4088 g_loss: 0.0284 (mse: 0.
0112, psnr: 26.7248, accuracy: 0.4844)
Epoch [ 5/20]    5 time: 64.66s, d_loss: 1.4341 g_loss: 0.0291 (mse: 0.
0103, psnr: 27.0913, accuracy: 0.5000)
Epoch [ 5/20]    6 time: 64.67s, d_loss: 1.4124 g_loss: 0.0331 (mse: 0.
0150, psnr: 26.3694, accuracy: 0.5156)
Epoch [ 5/20]    7 time: 64.71s, d_loss: 1.3869 g_loss: 0.0274 (mse: 0.
0125, psnr: 25.4349, accuracy: 0.5000)
Epoch [ 5/20]    8 time: 64.64s, d_loss: 1.4120 g_loss: 0.0297 (mse: 0.
0108, psnr: 26.4936, accuracy: 0.5156)
Epoch [ 5/20]    9 time: 68.09s, d_loss: 1.4131 g_loss: 0.0286 (mse: 0.
```

```
0098, psnr: 27.0650, accuracy: 0.5156)
Epoch [ 5/20]   10 time: 64.71s, d_loss: 1.3907 g_loss: 0.0262 (mse: 0.
0099, psnr: 27.1010, accuracy: 0.5156)
Epoch [ 5/20]   11 time: 64.55s, d_loss: 1.4080 g_loss: 0.0283 (mse: 0.
0093, psnr: 27.3589, accuracy: 0.5312)
Epoch [ 5/20]   12 time: 64.61s, d_loss: 1.3857 g_loss: 0.0309 (mse: 0.
0103, psnr: 26.5851, accuracy: 0.5312)
Epoch [ 5/20]   13 time: 64.66s, d_loss: 1.3806 g_loss: 0.0327 (mse: 0.
0133, psnr: 25.9350, accuracy: 0.5000)
Epoch [ 5/20]   14 time: 64.75s, d_loss: 1.4027 g_loss: 0.0303 (mse: 0.
0101, psnr: 26.6621, accuracy: 0.5156)
Epoch [ 5/20]   15 time: 64.67s, d_loss: 1.4139 g_loss: 0.0258 (mse: 0.
0101, psnr: 26.8313, accuracy: 0.5000)
Epoch [ 5/20]   16 time: 64.97s, d_loss: 1.4386 g_loss: 0.0337 (mse: 0.
0135, psnr: 25.7861, accuracy: 0.5156)
Epoch [ 5/20]   17 time: 64.52s, d_loss: 1.4325 g_loss: 0.0292 (mse: 0.
0108, psnr: 26.3518, accuracy: 0.5000)
Epoch [ 5/20]   18 time: 64.71s, d_loss: 1.4652 g_loss: 0.0362 (mse: 0.
0156, psnr: 25.8708, accuracy: 0.5312)
Epoch [ 5/20]   19 time: 64.76s, d_loss: 1.4486 g_loss: 0.0330 (mse: 0.
0135, psnr: 25.7352, accuracy: 0.5000)
Epoch [ 5/20]   20 time: 64.94s, d_loss: 1.4702 g_loss: 0.0335 (mse: 0.
0143, psnr: 25.9213, accuracy: 0.5000)
Epoch [ 5/20]   21 time: 64.60s, d_loss: 1.4166 g_loss: 0.0381 (mse: 0.
0183, psnr: 25.4464, accuracy: 0.5469)
Epoch [ 5/20]   22 time: 64.88s, d_loss: 1.4250 g_loss: 0.0298 (mse: 0.
0116, psnr: 25.8218, accuracy: 0.5000)
Epoch [ 5/20]   23 time: 64.90s, d_loss: 1.4039 g_loss: 0.0344 (mse: 0.
0127, psnr: 25.7000, accuracy: 0.4844)
Epoch [ 5/20]   24 time: 65.66s, d_loss: 1.4359 g_loss: 0.0288 (mse: 0.
0104, psnr: 26.7139, accuracy: 0.5000)
Epoch [ 5/20]   25 time: 65.04s, d_loss: 1.3808 g_loss: 0.0354 (mse: 0.
0133, psnr: 25.7678, accuracy: 0.5000)
Epoch [ 5/20]   26 time: 64.55s, d_loss: 1.4055 g_loss: 0.0307 (mse: 0.
0112, psnr: 26.5255, accuracy: 0.5312)
Epoch [ 5/20]   27 time: 64.56s, d_loss: 1.4412 g_loss: 0.0302 (mse: 0.
0130, psnr: 26.3554, accuracy: 0.5312)
Epoch [ 5/20]   28 time: 64.89s, d_loss: 1.4365 g_loss: 0.0319 (mse: 0.
0127, psnr: 25.7748, accuracy: 0.5156)
Epoch [ 5/20]   29 time: 64.75s, d_loss: 1.4212 g_loss: 0.0313 (mse: 0.
0113, psnr: 26.8144, accuracy: 0.5000)
Epoch [ 5/20]   30 time: 64.67s, d_loss: 1.4246 g_loss: 0.0243 (mse: 0.
0091, psnr: 27.4015, accuracy: 0.5000)
Epoch [ 5/20]   31 time: 64.87s, d_loss: 1.4077 g_loss: 0.0275 (mse: 0.
0095, psnr: 27.2261, accuracy: 0.5000)
Epoch [ 5/20]   32 time: 64.93s, d_loss: 1.4193 g_loss: 0.0289 (mse: 0.
0090, psnr: 27.3388, accuracy: 0.5000)
Epoch [ 5/20]   33 time: 64.66s, d_loss: 1.3865 g_loss: 0.0337 (mse: 0.
0138, psnr: 25.9294, accuracy: 0.5156)
Epoch [ 5/20]   34 time: 64.95s, d_loss: 1.3897 g_loss: 0.0328 (mse: 0.
0112, psnr: 26.4221, accuracy: 0.5312)
Epoch [ 5/20]   35 time: 64.62s, d_loss: 1.4321 g_loss: 0.0288 (mse: 0.
0096, psnr: 27.2471, accuracy: 0.5312)
Epoch [ 5/20]   36 time: 64.92s, d_loss: 1.4252 g_loss: 0.0353 (mse: 0.
0118, psnr: 26.1134, accuracy: 0.5000)
Epoch [ 5/20]   37 time: 65.07s, d_loss: 1.4041 g_loss: 0.0331 (mse: 0.
0124, psnr: 26.1303, accuracy: 0.5312)
```

```
Epoch [ 5/20]   38 time: 65.05s, d_loss: 1.4003 g_loss: 0.0332 (mse: 0.
0117, psnr: 25.9593, accuracy: 0.5000)
Epoch [ 5/20]   39 time: 64.96s, d_loss: 1.4916 g_loss: 0.0277 (mse: 0.
0101, psnr: 26.8592, accuracy: 0.5000)
Epoch [ 5/20]   40 time: 64.84s, d_loss: 1.4790 g_loss: 0.0298 (mse: 0.
0113, psnr: 26.5050, accuracy: 0.5625)
Epoch [ 5/20]   41 time: 65.00s, d_loss: 1.3802 g_loss: 0.0352 (mse: 0.
0143, psnr: 25.3841, accuracy: 0.5000)
Epoch [ 5/20]   42 time: 64.72s, d_loss: 1.4730 g_loss: 0.0303 (mse: 0.
0111, psnr: 26.4328, accuracy: 0.5000)
Epoch [ 5/20]   43 time: 64.99s, d_loss: 1.4514 g_loss: 0.0401 (mse: 0.
0166, psnr: 24.1479, accuracy: 0.5000)
Epoch [ 5/20]   44 time: 64.80s, d_loss: 1.3873 g_loss: 0.0259 (mse: 0.
0091, psnr: 27.2451, accuracy: 0.4844)
Epoch [ 5/20]   45 time: 65.10s, d_loss: 1.4297 g_loss: 0.0382 (mse: 0.
0189, psnr: 24.7827, accuracy: 0.5156)
Epoch [ 5/20]   46 time: 56.57s, d_loss: 1.4271 g_loss: 0.0367 (mse: 0.
0137, psnr: 25.6309, accuracy: 0.5179)
[*] Epoch: [ 5/20] time: 3040.45s, d_loss: 1.4204 g_loss: 0.0312 (mse:
 0.011940, psnr: 26.3218, accuracy: 0.5107)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 6/20]    0 time: 65.23s, d_loss: 1.4382 g_loss: 0.0235 (mse: 0.
0091, psnr: 27.3336, accuracy: 0.5312)
Epoch [ 6/20]    1 time: 64.65s, d_loss: 1.3481 g_loss: 0.0364 (mse: 0.
0180, psnr: 25.6172, accuracy: 0.5156)
Epoch [ 6/20]    2 time: 64.52s, d_loss: 1.3924 g_loss: 0.0305 (mse: 0.
0117, psnr: 26.4504, accuracy: 0.5312)
Epoch [ 6/20]    3 time: 64.62s, d_loss: 1.4146 g_loss: 0.0332 (mse: 0.
0123, psnr: 25.7670, accuracy: 0.4844)
Epoch [ 6/20]    4 time: 64.59s, d_loss: 1.4471 g_loss: 0.0363 (mse: 0.
0164, psnr: 24.6391, accuracy: 0.4688)
Epoch [ 6/20]    5 time: 64.51s, d_loss: 1.4426 g_loss: 0.0289 (mse: 0.
0101, psnr: 27.0208, accuracy: 0.5000)
Epoch [ 6/20]    6 time: 64.42s, d_loss: 1.5092 g_loss: 0.0268 (mse: 0.
0096, psnr: 26.7896, accuracy: 0.5000)
Epoch [ 6/20]    7 time: 64.63s, d_loss: 1.4272 g_loss: 0.0306 (mse: 0.
0112, psnr: 26.2337, accuracy: 0.5625)
Epoch [ 6/20]    8 time: 64.85s, d_loss: 1.4154 g_loss: 0.0269 (mse: 0.
0086, psnr: 27.4275, accuracy: 0.5312)
Epoch [ 6/20]    9 time: 64.54s, d_loss: 1.4055 g_loss: 0.0268 (mse: 0.
0090, psnr: 27.5807, accuracy: 0.5469)
Epoch [ 6/20]   10 time: 64.73s, d_loss: 1.4448 g_loss: 0.0295 (mse: 0.
0125, psnr: 26.9649, accuracy: 0.5625)
Epoch [ 6/20]   11 time: 64.59s, d_loss: 1.4101 g_loss: 0.0254 (mse: 0.
0086, psnr: 27.8449, accuracy: 0.5000)
Epoch [ 6/20]   12 time: 64.53s, d_loss: 1.4058 g_loss: 0.0339 (mse: 0.
0132, psnr: 25.6705, accuracy: 0.5469)
Epoch [ 6/20]   13 time: 64.70s, d_loss: 1.4449 g_loss: 0.0334 (mse: 0.
0131, psnr: 26.0130, accuracy: 0.5000)
Epoch [ 6/20]   14 time: 64.72s, d_loss: 1.4416 g_loss: 0.0324 (mse: 0.
0122, psnr: 26.0900, accuracy: 0.4844)
Epoch [ 6/20]   15 time: 64.70s, d_loss: 1.5113 g_loss: 0.0335 (mse: 0.
0120, psnr: 26.0422, accuracy: 0.5000)
```

```
Epoch [ 6/20]   16 time: 64.72s, d_loss: 1.3922 g_loss: 0.0290 (mse: 0.
0114, psnr: 26.4065, accuracy: 0.5156)
Epoch [ 6/20]   17 time: 64.85s, d_loss: 1.4509 g_loss: 0.0324 (mse: 0.
0116, psnr: 26.2560, accuracy: 0.5000)
Epoch [ 6/20]   18 time: 64.40s, d_loss: 1.4684 g_loss: 0.0339 (mse: 0.
0152, psnr: 25.6232, accuracy: 0.5000)
Epoch [ 6/20]   19 time: 64.32s, d_loss: 1.4107 g_loss: 0.0301 (mse: 0.
0123, psnr: 26.3259, accuracy: 0.5469)
Epoch [ 6/20]   20 time: 64.42s, d_loss: 1.4415 g_loss: 0.0212 (mse: 0.
0075, psnr: 27.9999, accuracy: 0.5312)
Epoch [ 6/20]   21 time: 64.59s, d_loss: 1.4089 g_loss: 0.0282 (mse: 0.
0094, psnr: 27.0494, accuracy: 0.5000)
Epoch [ 6/20]   22 time: 64.84s, d_loss: 1.6696 g_loss: 0.0285 (mse: 0.
0118, psnr: 26.0980, accuracy: 0.5000)
Epoch [ 6/20]   23 time: 65.03s, d_loss: 1.4446 g_loss: 0.0334 (mse: 0.
0131, psnr: 25.1885, accuracy: 0.5312)
Epoch [ 6/20]   24 time: 64.89s, d_loss: 1.5216 g_loss: 0.0297 (mse: 0.
0119, psnr: 25.9360, accuracy: 0.4531)
Epoch [ 6/20]   25 time: 64.50s, d_loss: 1.4895 g_loss: 0.0385 (mse: 0.
0172, psnr: 25.4273, accuracy: 0.5625)
Epoch [ 6/20]   26 time: 64.61s, d_loss: 1.4340 g_loss: 0.0298 (mse: 0.
0105, psnr: 26.6515, accuracy: 0.5312)
Epoch [ 6/20]   27 time: 65.22s, d_loss: 1.4472 g_loss: 0.0345 (mse: 0.
0131, psnr: 25.8151, accuracy: 0.5000)
Epoch [ 6/20]   28 time: 64.72s, d_loss: 1.4594 g_loss: 0.0336 (mse: 0.
0172, psnr: 25.9769, accuracy: 0.5312)
Epoch [ 6/20]   29 time: 64.54s, d_loss: 1.5282 g_loss: 0.0265 (mse: 0.
0092, psnr: 27.2283, accuracy: 0.5000)
Epoch [ 6/20]   30 time: 64.52s, d_loss: 1.4838 g_loss: 0.0328 (mse: 0.
0118, psnr: 26.7528, accuracy: 0.5000)
Epoch [ 6/20]   31 time: 64.51s, d_loss: 1.4828 g_loss: 0.0344 (mse: 0.
0153, psnr: 24.8165, accuracy: 0.5000)
Epoch [ 6/20]   32 time: 64.89s, d_loss: 1.5560 g_loss: 0.0319 (mse: 0.
0118, psnr: 26.5871, accuracy: 0.4844)
Epoch [ 6/20]   33 time: 64.41s, d_loss: 1.5282 g_loss: 0.0305 (mse: 0.
0112, psnr: 26.7310, accuracy: 0.5000)
Epoch [ 6/20]   34 time: 64.44s, d_loss: 1.4821 g_loss: 0.0307 (mse: 0.
0125, psnr: 25.5897, accuracy: 0.4688)
Epoch [ 6/20]   35 time: 64.45s, d_loss: 1.4198 g_loss: 0.0277 (mse: 0.
0104, psnr: 26.7366, accuracy: 0.5156)
Epoch [ 6/20]   36 time: 64.89s, d_loss: 1.4442 g_loss: 0.0335 (mse: 0.
0163, psnr: 26.6455, accuracy: 0.5156)
Epoch [ 6/20]   37 time: 68.70s, d_loss: 1.5208 g_loss: 0.0292 (mse: 0.
0116, psnr: 26.3931, accuracy: 0.5000)
Epoch [ 6/20]   38 time: 66.49s, d_loss: 1.4918 g_loss: 0.0284 (mse: 0.
0093, psnr: 27.3662, accuracy: 0.5312)
Epoch [ 6/20]   39 time: 64.55s, d_loss: 1.4182 g_loss: 0.0307 (mse: 0.
0108, psnr: 26.3944, accuracy: 0.4688)
Epoch [ 6/20]   40 time: 64.71s, d_loss: 1.4675 g_loss: 0.0279 (mse: 0.
0109, psnr: 26.8298, accuracy: 0.5312)
Epoch [ 6/20]   41 time: 64.55s, d_loss: 1.5010 g_loss: 0.0311 (mse: 0.
0103, psnr: 26.8099, accuracy: 0.5156)
Epoch [ 6/20]   42 time: 64.72s, d_loss: 1.4327 g_loss: 0.0336 (mse: 0.
0140, psnr: 26.0101, accuracy: 0.5156)
Epoch [ 6/20]   43 time: 64.71s, d_loss: 1.4240 g_loss: 0.0262 (mse: 0.
0085, psnr: 27.7601, accuracy: 0.5000)
Epoch [ 6/20]   44 time: 65.04s, d_loss: 1.4553 g_loss: 0.0290 (mse: 0.
```

```
0116, psnr: 26.7907, accuracy: 0.4844)
Epoch [ 6/20]   45 time: 64.85s, d_loss: 1.4919 g_loss: 0.0267 (mse: 0.
0105, psnr: 26.5321, accuracy: 0.5000)
Epoch [ 6/20]   46 time: 56.73s, d_loss: 1.4395 g_loss: 0.0277 (mse: 0.
0123, psnr: 26.4737, accuracy: 0.5000)
[*] Epoch: [ 6/20] time: 3037.38s, d_loss: 1.4576 g_loss: 0.0304 (mse:
 0.011876, psnr: 26.4401, accuracy: 0.5106)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 7/20]    0 time: 66.62s, d_loss: 1.3712 g_loss: 0.0262 (mse: 0.
0118, psnr: 26.7954, accuracy: 0.5156)
Epoch [ 7/20]    1 time: 65.12s, d_loss: 1.4620 g_loss: 0.0292 (mse: 0.
0106, psnr: 26.4641, accuracy: 0.5000)
Epoch [ 7/20]    2 time: 68.61s, d_loss: 1.4858 g_loss: 0.0305 (mse: 0.
0109, psnr: 26.6165, accuracy: 0.4688)
Epoch [ 7/20]    3 time: 68.12s, d_loss: 1.4245 g_loss: 0.0293 (mse: 0.
0098, psnr: 27.1104, accuracy: 0.5000)
Epoch [ 7/20]    4 time: 68.35s, d_loss: 1.3614 g_loss: 0.0388 (mse: 0.
0201, psnr: 25.1845, accuracy: 0.5781)
Epoch [ 7/20]    5 time: 67.36s, d_loss: 1.3949 g_loss: 0.0251 (mse: 0.
0082, psnr: 27.7067, accuracy: 0.5156)
Epoch [ 7/20]    6 time: 66.86s, d_loss: 1.4317 g_loss: 0.0359 (mse: 0.
0139, psnr: 25.4152, accuracy: 0.5156)
Epoch [ 7/20]    7 time: 67.93s, d_loss: 1.4366 g_loss: 0.0279 (mse: 0.
0105, psnr: 26.5559, accuracy: 0.5000)
Epoch [ 7/20]    8 time: 67.88s, d_loss: 1.3919 g_loss: 0.0265 (mse: 0.
0090, psnr: 27.4069, accuracy: 0.5000)
Epoch [ 7/20]    9 time: 64.78s, d_loss: 1.4328 g_loss: 0.0280 (mse: 0.
0103, psnr: 26.3303, accuracy: 0.5000)
Epoch [ 7/20]   10 time: 64.73s, d_loss: 1.3884 g_loss: 0.0275 (mse: 0.
0106, psnr: 26.3817, accuracy: 0.5000)
Epoch [ 7/20]   11 time: 64.93s, d_loss: 1.3912 g_loss: 0.0307 (mse: 0.
0116, psnr: 26.4757, accuracy: 0.5000)
Epoch [ 7/20]   12 time: 65.04s, d_loss: 1.3868 g_loss: 0.0325 (mse: 0.
0114, psnr: 26.4401, accuracy: 0.5156)
Epoch [ 7/20]   13 time: 64.74s, d_loss: 1.3886 g_loss: 0.0278 (mse: 0.
0097, psnr: 27.0303, accuracy: 0.5000)
Epoch [ 7/20]   14 time: 64.77s, d_loss: 1.3933 g_loss: 0.0298 (mse: 0.
0117, psnr: 26.1283, accuracy: 0.5000)
Epoch [ 7/20]   15 time: 64.89s, d_loss: 1.3905 g_loss: 0.0348 (mse: 0.
0149, psnr: 26.0213, accuracy: 0.5156)
Epoch [ 7/20]   16 time: 64.84s, d_loss: 1.3996 g_loss: 0.0316 (mse: 0.
0132, psnr: 26.3082, accuracy: 0.4844)
Epoch [ 7/20]   17 time: 64.94s, d_loss: 1.3837 g_loss: 0.0194 (mse: 0.
0071, psnr: 28.2132, accuracy: 0.5312)
Epoch [ 7/20]   18 time: 65.07s, d_loss: 1.4046 g_loss: 0.0252 (mse: 0.
0086, psnr: 27.2910, accuracy: 0.5000)
Epoch [ 7/20]   19 time: 64.80s, d_loss: 1.3915 g_loss: 0.0294 (mse: 0.
0120, psnr: 26.4865, accuracy: 0.5000)
Epoch [ 7/20]   20 time: 64.82s, d_loss: 1.3888 g_loss: 0.0308 (mse: 0.
0128, psnr: 26.0959, accuracy: 0.5000)
Epoch [ 7/20]   21 time: 64.77s, d_loss: 1.4022 g_loss: 0.0274 (mse: 0.
0117, psnr: 25.8186, accuracy: 0.5000)
Epoch [ 7/20]   22 time: 65.00s, d_loss: 1.4210 g_loss: 0.0333 (mse: 0.
```

```
0125, psnr: 26.2438, accuracy: 0.5156)
Epoch [ 7/20]   23 time: 65.30s, d_loss: 1.3682 g_loss: 0.0302 (mse: 0.
0121, psnr: 26.6700, accuracy: 0.5156)
Epoch [ 7/20]   24 time: 65.19s, d_loss: 1.3822 g_loss: 0.0291 (mse: 0.
0117, psnr: 26.5971, accuracy: 0.5156)
Epoch [ 7/20]   25 time: 64.84s, d_loss: 1.4072 g_loss: 0.0350 (mse: 0.
0132, psnr: 25.4043, accuracy: 0.5156)
Epoch [ 7/20]   26 time: 65.54s, d_loss: 1.3914 g_loss: 0.0290 (mse: 0.
0101, psnr: 27.3528, accuracy: 0.5469)
Epoch [ 7/20]   27 time: 64.56s, d_loss: 1.3823 g_loss: 0.0246 (mse: 0.
0087, psnr: 27.8433, accuracy: 0.5312)
Epoch [ 7/20]   28 time: 66.19s, d_loss: 1.3913 g_loss: 0.0292 (mse: 0.
0131, psnr: 26.5318, accuracy: 0.5000)
Epoch [ 7/20]   29 time: 64.82s, d_loss: 1.3741 g_loss: 0.0254 (mse: 0.
0091, psnr: 26.9510, accuracy: 0.5156)
Epoch [ 7/20]   30 time: 64.52s, d_loss: 1.3928 g_loss: 0.0276 (mse: 0.
0114, psnr: 26.3212, accuracy: 0.5000)
Epoch [ 7/20]   31 time: 64.59s, d_loss: 1.3906 g_loss: 0.0275 (mse: 0.
0094, psnr: 27.1420, accuracy: 0.5000)
Epoch [ 7/20]   32 time: 64.99s, d_loss: 1.3863 g_loss: 0.0372 (mse: 0.
0134, psnr: 25.2819, accuracy: 0.5000)
Epoch [ 7/20]   33 time: 65.43s, d_loss: 1.3995 g_loss: 0.0364 (mse: 0.
0123, psnr: 26.4785, accuracy: 0.5000)
Epoch [ 7/20]   34 time: 65.37s, d_loss: 1.4226 g_loss: 0.0271 (mse: 0.
0099, psnr: 26.9036, accuracy: 0.5156)
Epoch [ 7/20]   35 time: 64.92s, d_loss: 1.4113 g_loss: 0.0285 (mse: 0.
0101, psnr: 26.5307, accuracy: 0.5000)
Epoch [ 7/20]   36 time: 65.51s, d_loss: 1.3908 g_loss: 0.0279 (mse: 0.
0106, psnr: 26.7747, accuracy: 0.4844)
Epoch [ 7/20]   37 time: 64.63s, d_loss: 1.3881 g_loss: 0.0252 (mse: 0.
0096, psnr: 27.0098, accuracy: 0.4844)
Epoch [ 7/20]   38 time: 64.56s, d_loss: 1.3622 g_loss: 0.0299 (mse: 0.
0101, psnr: 26.9118, accuracy: 0.5000)
Epoch [ 7/20]   39 time: 64.59s, d_loss: 1.3821 g_loss: 0.0245 (mse: 0.
0094, psnr: 26.9577, accuracy: 0.5156)
Epoch [ 7/20]   40 time: 64.61s, d_loss: 1.3574 g_loss: 0.0288 (mse: 0.
0108, psnr: 26.5722, accuracy: 0.5469)
Epoch [ 7/20]   41 time: 64.68s, d_loss: 1.3712 g_loss: 0.0291 (mse: 0.
0099, psnr: 26.8711, accuracy: 0.5156)
Epoch [ 7/20]   42 time: 65.18s, d_loss: 1.3674 g_loss: 0.0284 (mse: 0.
0109, psnr: 26.9304, accuracy: 0.5000)
Epoch [ 7/20]   43 time: 64.94s, d_loss: 1.3945 g_loss: 0.0250 (mse: 0.
0080, psnr: 27.8048, accuracy: 0.5156)
Epoch [ 7/20]   44 time: 64.63s, d_loss: 1.3558 g_loss: 0.0292 (mse: 0.
0122, psnr: 25.6577, accuracy: 0.5312)
Epoch [ 7/20]   45 time: 65.11s, d_loss: 1.3800 g_loss: 0.0270 (mse: 0.
0115, psnr: 26.6862, accuracy: 0.5156)
Epoch [ 7/20]   46 time: 56.81s, d_loss: 1.4075 g_loss: 0.0264 (mse: 0.
0095, psnr: 27.0070, accuracy: 0.5714)
[*] Epoch: [ 7/20] time: 3066.49s, d_loss: 1.3953 g_loss: 0.0291 (mse:
 0.011053, psnr: 26.6322, accuracy: 0.5105)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 8/20]    0 time: 64.62s, d_loss: 1.3692 g_loss: 0.0324 (mse: 0.
```

```
0160, psnr: 25.2976, accuracy: 0.5156)
Epoch [ 8/20]    1 time: 64.93s, d_loss: 1.3938 g_loss: 0.0226 (mse: 0.
0076, psnr: 28.4130, accuracy: 0.4844)
Epoch [ 8/20]    2 time: 64.36s, d_loss: 1.4087 g_loss: 0.0282 (mse: 0.
0109, psnr: 26.2927, accuracy: 0.5000)
Epoch [ 8/20]    3 time: 64.54s, d_loss: 1.3966 g_loss: 0.0282 (mse: 0.
0099, psnr: 26.8826, accuracy: 0.4844)
Epoch [ 8/20]    4 time: 64.61s, d_loss: 1.3601 g_loss: 0.0280 (mse: 0.
0101, psnr: 26.9308, accuracy: 0.5312)
Epoch [ 8/20]    5 time: 64.19s, d_loss: 1.3728 g_loss: 0.0284 (mse: 0.
0099, psnr: 27.0950, accuracy: 0.5312)
Epoch [ 8/20]    6 time: 64.54s, d_loss: 1.3769 g_loss: 0.0279 (mse: 0.
0112, psnr: 26.5167, accuracy: 0.5312)
Epoch [ 8/20]    7 time: 64.61s, d_loss: 1.3596 g_loss: 0.0246 (mse: 0.
0093, psnr: 27.2880, accuracy: 0.5312)
Epoch [ 8/20]    8 time: 64.66s, d_loss: 1.3740 g_loss: 0.0300 (mse: 0.
0111, psnr: 26.6048, accuracy: 0.5156)
Epoch [ 8/20]    9 time: 64.66s, d_loss: 1.4461 g_loss: 0.0308 (mse: 0.
0141, psnr: 27.2406, accuracy: 0.5000)
Epoch [ 8/20]   10 time: 64.64s, d_loss: 1.4184 g_loss: 0.0224 (mse: 0.
0075, psnr: 28.3138, accuracy: 0.5312)
Epoch [ 8/20]   11 time: 64.33s, d_loss: 1.3697 g_loss: 0.0277 (mse: 0.
0095, psnr: 26.8978, accuracy: 0.5469)
Epoch [ 8/20]   12 time: 64.57s, d_loss: 1.3967 g_loss: 0.0271 (mse: 0.
0094, psnr: 27.3140, accuracy: 0.5000)
Epoch [ 8/20]   13 time: 64.46s, d_loss: 1.3547 g_loss: 0.0277 (mse: 0.
0106, psnr: 26.6376, accuracy: 0.5156)
Epoch [ 8/20]   14 time: 64.62s, d_loss: 1.4263 g_loss: 0.0313 (mse: 0.
0117, psnr: 25.8825, accuracy: 0.5000)
Epoch [ 8/20]   15 time: 64.72s, d_loss: 1.4514 g_loss: 0.0284 (mse: 0.
0111, psnr: 26.0897, accuracy: 0.5156)
Epoch [ 8/20]   16 time: 64.53s, d_loss: 1.4032 g_loss: 0.0227 (mse: 0.
0079, psnr: 27.7622, accuracy: 0.5156)
Epoch [ 8/20]   17 time: 64.39s, d_loss: 1.3621 g_loss: 0.0330 (mse: 0.
0113, psnr: 26.1033, accuracy: 0.5312)
Epoch [ 8/20]   18 time: 64.52s, d_loss: 1.3819 g_loss: 0.0319 (mse: 0.
0120, psnr: 25.8250, accuracy: 0.5000)
Epoch [ 8/20]   19 time: 64.57s, d_loss: 1.3188 g_loss: 0.0296 (mse: 0.
0115, psnr: 26.6218, accuracy: 0.5469)
Epoch [ 8/20]   20 time: 64.37s, d_loss: 1.4377 g_loss: 0.0215 (mse: 0.
0076, psnr: 28.0191, accuracy: 0.5625)
Epoch [ 8/20]   21 time: 64.29s, d_loss: 1.3735 g_loss: 0.0283 (mse: 0.
0102, psnr: 26.8588, accuracy: 0.5625)
Epoch [ 8/20]   22 time: 64.43s, d_loss: 1.3854 g_loss: 0.0298 (mse: 0.
0106, psnr: 26.7550, accuracy: 0.4844)
Epoch [ 8/20]   23 time: 64.38s, d_loss: 1.3802 g_loss: 0.0341 (mse: 0.
0130, psnr: 25.9898, accuracy: 0.5000)
Epoch [ 8/20]   24 time: 64.66s, d_loss: 1.4116 g_loss: 0.0265 (mse: 0.
0091, psnr: 27.0987, accuracy: 0.5000)
Epoch [ 8/20]   25 time: 64.53s, d_loss: 1.4304 g_loss: 0.0273 (mse: 0.
0107, psnr: 26.4841, accuracy: 0.5000)
Epoch [ 8/20]   26 time: 64.45s, d_loss: 1.4216 g_loss: 0.0330 (mse: 0.
0126, psnr: 25.6104, accuracy: 0.5312)
Epoch [ 8/20]   27 time: 64.59s, d_loss: 1.4519 g_loss: 0.0361 (mse: 0.
0140, psnr: 25.9122, accuracy: 0.5625)
Epoch [ 8/20]   28 time: 65.07s, d_loss: 1.4244 g_loss: 0.0295 (mse: 0.
0097, psnr: 26.9795, accuracy: 0.5156)
```

```
Epoch [ 8/20]   29 time: 64.45s, d_loss: 1.3771 g_loss: 0.0250 (mse: 0.
0100, psnr: 27.0826, accuracy: 0.5156)
Epoch [ 8/20]   30 time: 64.45s, d_loss: 1.4948 g_loss: 0.0265 (mse: 0.
0097, psnr: 26.8696, accuracy: 0.5000)
Epoch [ 8/20]   31 time: 64.57s, d_loss: 1.4014 g_loss: 0.0331 (mse: 0.
0142, psnr: 25.8101, accuracy: 0.5156)
Epoch [ 8/20]   32 time: 64.52s, d_loss: 1.3406 g_loss: 0.0307 (mse: 0.
0138, psnr: 25.9928, accuracy: 0.5156)
Epoch [ 8/20]   33 time: 64.74s, d_loss: 1.4784 g_loss: 0.0318 (mse: 0.
0121, psnr: 26.1992, accuracy: 0.5000)
Epoch [ 8/20]   34 time: 64.49s, d_loss: 1.4514 g_loss: 0.0304 (mse: 0.
0121, psnr: 25.7687, accuracy: 0.5312)
Epoch [ 8/20]   35 time: 64.44s, d_loss: 1.3722 g_loss: 0.0300 (mse: 0.
0120, psnr: 26.0534, accuracy: 0.5156)
Epoch [ 8/20]   36 time: 64.51s, d_loss: 1.3886 g_loss: 0.0264 (mse: 0.
0119, psnr: 25.9207, accuracy: 0.5156)
Epoch [ 8/20]   37 time: 64.98s, d_loss: 1.3758 g_loss: 0.0260 (mse: 0.
0096, psnr: 27.7127, accuracy: 0.5312)
Epoch [ 8/20]   38 time: 64.31s, d_loss: 1.3797 g_loss: 0.0308 (mse: 0.
0128, psnr: 26.2528, accuracy: 0.5156)
Epoch [ 8/20]   39 time: 64.60s, d_loss: 1.3515 g_loss: 0.0354 (mse: 0.
0129, psnr: 25.9379, accuracy: 0.5469)
Epoch [ 8/20]   40 time: 64.42s, d_loss: 1.3488 g_loss: 0.0260 (mse: 0.
0099, psnr: 27.2951, accuracy: 0.5312)
Epoch [ 8/20]   41 time: 64.80s, d_loss: 1.3407 g_loss: 0.0300 (mse: 0.
0114, psnr: 26.6419, accuracy: 0.5156)
Epoch [ 8/20]   42 time: 64.67s, d_loss: 1.4014 g_loss: 0.0320 (mse: 0.
0119, psnr: 26.1624, accuracy: 0.5000)
Epoch [ 8/20]   43 time: 64.63s, d_loss: 1.3874 g_loss: 0.0323 (mse: 0.
0110, psnr: 26.8411, accuracy: 0.5469)
Epoch [ 8/20]   44 time: 64.64s, d_loss: 1.4518 g_loss: 0.0345 (mse: 0.
0126, psnr: 25.5726, accuracy: 0.5000)
Epoch [ 8/20]   45 time: 64.73s, d_loss: 1.3799 g_loss: 0.0385 (mse: 0.
0175, psnr: 24.6415, accuracy: 0.5469)
Epoch [ 8/20]   46 time: 56.61s, d_loss: 1.3938 g_loss: 0.0300 (mse: 0.
0136, psnr: 26.1932, accuracy: 0.5179)
[*] Epoch: [ 8/20] time: 3026.41s, d_loss: 1.3952 g_loss: 0.0293 (mse:
 0.011253, psnr: 26.5673, accuracy: 0.5193)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [ 9/20]    0 time: 64.55s, d_loss: 1.3293 g_loss: 0.0282 (mse: 0.
0127, psnr: 25.9925, accuracy: 0.5469)
Epoch [ 9/20]    1 time: 64.55s, d_loss: 1.3452 g_loss: 0.0223 (mse: 0.
0077, psnr: 28.1198, accuracy: 0.5469)
Epoch [ 9/20]    2 time: 64.48s, d_loss: 1.4048 g_loss: 0.0265 (mse: 0.
0103, psnr: 26.7473, accuracy: 0.5000)
Epoch [ 9/20]    3 time: 64.56s, d_loss: 1.4538 g_loss: 0.0303 (mse: 0.
0121, psnr: 26.3013, accuracy: 0.5000)
Epoch [ 9/20]    4 time: 64.50s, d_loss: 1.3649 g_loss: 0.0331 (mse: 0.
0125, psnr: 26.3199, accuracy: 0.5938)
Epoch [ 9/20]    5 time: 64.56s, d_loss: 1.3543 g_loss: 0.0340 (mse: 0.
0126, psnr: 25.9105, accuracy: 0.5625)
Epoch [ 9/20]    6 time: 64.64s, d_loss: 1.3848 g_loss: 0.0244 (mse: 0.
0083, psnr: 27.6096, accuracy: 0.5312)
```

```
Epoch [ 9/20]    7 time: 64.66s, d_loss: 1.3740 g_loss: 0.0306 (mse: 0.
0122, psnr: 26.1271, accuracy: 0.5000)
Epoch [ 9/20]    8 time: 64.63s, d_loss: 1.3454 g_loss: 0.0247 (mse: 0.
0109, psnr: 27.8149, accuracy: 0.5312)
Epoch [ 9/20]    9 time: 64.36s, d_loss: 1.3760 g_loss: 0.0245 (mse: 0.
0092, psnr: 26.9797, accuracy: 0.5156)
Epoch [ 9/20]   10 time: 64.35s, d_loss: 1.4191 g_loss: 0.0327 (mse: 0.
0144, psnr: 26.3838, accuracy: 0.5156)
Epoch [ 9/20]   11 time: 64.51s, d_loss: 1.3437 g_loss: 0.0295 (mse: 0.
0118, psnr: 26.3110, accuracy: 0.5156)
Epoch [ 9/20]   12 time: 65.55s, d_loss: 1.3933 g_loss: 0.0304 (mse: 0.
0106, psnr: 26.5909, accuracy: 0.5625)
Epoch [ 9/20]   13 time: 64.61s, d_loss: 1.3665 g_loss: 0.0309 (mse: 0.
0134, psnr: 26.4331, accuracy: 0.5156)
Epoch [ 9/20]   14 time: 64.45s, d_loss: 1.3554 g_loss: 0.0338 (mse: 0.
0169, psnr: 26.5007, accuracy: 0.5312)
Epoch [ 9/20]   15 time: 64.37s, d_loss: 1.3343 g_loss: 0.0335 (mse: 0.
0113, psnr: 26.4525, accuracy: 0.5156)
Epoch [ 9/20]   16 time: 64.49s, d_loss: 1.4203 g_loss: 0.0239 (mse: 0.
0087, psnr: 27.1666, accuracy: 0.5000)
Epoch [ 9/20]   17 time: 64.59s, d_loss: 1.4382 g_loss: 0.0274 (mse: 0.
0103, psnr: 26.4946, accuracy: 0.5625)
Epoch [ 9/20]   18 time: 64.51s, d_loss: 1.3188 g_loss: 0.0345 (mse: 0.
0131, psnr: 25.4544, accuracy: 0.6875)
Epoch [ 9/20]   19 time: 64.48s, d_loss: 1.3618 g_loss: 0.0302 (mse: 0.
0121, psnr: 26.2056, accuracy: 0.5312)
Epoch [ 9/20]   20 time: 64.28s, d_loss: 1.3417 g_loss: 0.0246 (mse: 0.
0089, psnr: 27.7653, accuracy: 0.5156)
Epoch [ 9/20]   21 time: 64.59s, d_loss: 1.3375 g_loss: 0.0237 (mse: 0.
0081, psnr: 27.8169, accuracy: 0.5312)
Epoch [ 9/20]   22 time: 64.35s, d_loss: 1.3341 g_loss: 0.0448 (mse: 0.
0266, psnr: 25.9135, accuracy: 0.5156)
Epoch [ 9/20]   23 time: 64.27s, d_loss: 1.3937 g_loss: 0.0239 (mse: 0.
0094, psnr: 26.8951, accuracy: 0.5000)
Epoch [ 9/20]   24 time: 64.51s, d_loss: 1.3939 g_loss: 0.0305 (mse: 0.
0109, psnr: 26.3251, accuracy: 0.5469)
Epoch [ 9/20]   25 time: 64.69s, d_loss: 1.3356 g_loss: 0.0308 (mse: 0.
0122, psnr: 25.8587, accuracy: 0.6250)
Epoch [ 9/20]   26 time: 64.33s, d_loss: 1.4373 g_loss: 0.0278 (mse: 0.
0098, psnr: 26.7528, accuracy: 0.5000)
Epoch [ 9/20]   27 time: 64.71s, d_loss: 1.4097 g_loss: 0.0267 (mse: 0.
0095, psnr: 26.7516, accuracy: 0.5000)
Epoch [ 9/20]   28 time: 64.40s, d_loss: 1.3982 g_loss: 0.0268 (mse: 0.
0094, psnr: 27.3008, accuracy: 0.6250)
Epoch [ 9/20]   29 time: 64.25s, d_loss: 1.5211 g_loss: 0.0251 (mse: 0.
0095, psnr: 27.4350, accuracy: 0.5469)
Epoch [ 9/20]   30 time: 64.52s, d_loss: 1.3441 g_loss: 0.0317 (mse: 0.
0112, psnr: 26.2026, accuracy: 0.5156)
Epoch [ 9/20]   31 time: 64.25s, d_loss: 1.3476 g_loss: 0.0259 (mse: 0.
0093, psnr: 26.9645, accuracy: 0.5000)
Epoch [ 9/20]   32 time: 64.34s, d_loss: 1.5926 g_loss: 0.0328 (mse: 0.
0144, psnr: 25.4300, accuracy: 0.5000)
Epoch [ 9/20]   33 time: 64.24s, d_loss: 1.3761 g_loss: 0.0294 (mse: 0.
0104, psnr: 26.5451, accuracy: 0.5312)
Epoch [ 9/20]   34 time: 64.40s, d_loss: 1.3881 g_loss: 0.0265 (mse: 0.
0108, psnr: 26.7486, accuracy: 0.5781)
Epoch [ 9/20]   35 time: 64.29s, d_loss: 1.4292 g_loss: 0.0292 (mse: 0.
```

```
0099, psnr: 26.8572, accuracy: 0.5469)
Epoch [ 9/20]   36 time: 64.52s, d_loss: 1.5231 g_loss: 0.0352 (mse: 0.
0175, psnr: 25.3277, accuracy: 0.6094)
Epoch [ 9/20]   37 time: 64.46s, d_loss: 1.2922 g_loss: 0.0323 (mse: 0.
0141, psnr: 26.7150, accuracy: 0.5469)
Epoch [ 9/20]   38 time: 64.45s, d_loss: 1.5357 g_loss: 0.0284 (mse: 0.
0120, psnr: 26.6983, accuracy: 0.5000)
Epoch [ 9/20]   39 time: 64.37s, d_loss: 1.4201 g_loss: 0.0211 (mse: 0.
0077, psnr: 27.7239, accuracy: 0.5000)
Epoch [ 9/20]   40 time: 64.39s, d_loss: 1.3767 g_loss: 0.0328 (mse: 0.
0146, psnr: 26.1187, accuracy: 0.5312)
Epoch [ 9/20]   41 time: 64.24s, d_loss: 1.3694 g_loss: 0.0306 (mse: 0.
0109, psnr: 26.9168, accuracy: 0.6094)
Epoch [ 9/20]   42 time: 64.40s, d_loss: 1.3947 g_loss: 0.0376 (mse: 0.
0132, psnr: 25.6770, accuracy: 0.5312)
Epoch [ 9/20]   43 time: 64.66s, d_loss: 1.4765 g_loss: 0.0205 (mse: 0.
0079, psnr: 28.6947, accuracy: 0.5000)
Epoch [ 9/20]   44 time: 64.15s, d_loss: 1.3777 g_loss: 0.0291 (mse: 0.
0099, psnr: 27.1298, accuracy: 0.5156)
Epoch [ 9/20]   45 time: 64.37s, d_loss: 1.4108 g_loss: 0.0207 (mse: 0.
0078, psnr: 28.3912, accuracy: 0.5156)
Epoch [ 9/20]   46 time: 56.37s, d_loss: 1.5240 g_loss: 0.0272 (mse: 0.
0098, psnr: 27.1202, accuracy: 0.5000)
[*] Epoch: [ 9/20] time: 3022.21s, d_loss: 1.3950 g_loss: 0.0290 (mse:
 0.011423, psnr: 26.7232, accuracy: 0.5362)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
 ** new learning rate: 0.000010 (for GAN)
Epoch [10/20]    0 time: 68.17s, d_loss: 1.3393 g_loss: 0.0260 (mse: 0.
0094, psnr: 27.4356, accuracy: 0.5312)
Epoch [10/20]    1 time: 66.92s, d_loss: 1.3414 g_loss: 0.0301 (mse: 0.
0136, psnr: 25.7439, accuracy: 0.5000)
Epoch [10/20]    2 time: 66.49s, d_loss: 1.3584 g_loss: 0.0253 (mse: 0.
0090, psnr: 27.1594, accuracy: 0.5000)
Epoch [10/20]    3 time: 64.90s, d_loss: 1.3423 g_loss: 0.0260 (mse: 0.
0093, psnr: 27.8079, accuracy: 0.5156)
Epoch [10/20]    4 time: 65.96s, d_loss: 1.3538 g_loss: 0.0295 (mse: 0.
0107, psnr: 26.7258, accuracy: 0.5000)
Epoch [10/20]    5 time: 65.86s, d_loss: 1.3439 g_loss: 0.0273 (mse: 0.
0137, psnr: 27.0276, accuracy: 0.5000)
Epoch [10/20]    6 time: 64.48s, d_loss: 1.3570 g_loss: 0.0316 (mse: 0.
0106, psnr: 26.8212, accuracy: 0.5156)
Epoch [10/20]    7 time: 64.47s, d_loss: 1.3570 g_loss: 0.0294 (mse: 0.
0099, psnr: 27.3368, accuracy: 0.5156)
Epoch [10/20]    8 time: 64.49s, d_loss: 1.4004 g_loss: 0.0242 (mse: 0.
0090, psnr: 27.5608, accuracy: 0.5156)
Epoch [10/20]    9 time: 64.63s, d_loss: 1.3982 g_loss: 0.0319 (mse: 0.
0113, psnr: 26.4326, accuracy: 0.6250)
Epoch [10/20]   10 time: 64.92s, d_loss: 1.3404 g_loss: 0.0226 (mse: 0.
0082, psnr: 27.8127, accuracy: 0.5000)
Epoch [10/20]   11 time: 66.79s, d_loss: 1.3813 g_loss: 0.0285 (mse: 0.
0097, psnr: 27.1032, accuracy: 0.5156)
Epoch [10/20]   12 time: 65.11s, d_loss: 1.3369 g_loss: 0.0244 (mse: 0.
0087, psnr: 27.5574, accuracy: 0.5156)
```

```
Epoch [10/20]   13 time: 65.98s, d_loss: 1.3016 g_loss: 0.0309 (mse: 0.
0122, psnr: 26.3522, accuracy: 0.5000)
Epoch [10/20]   14 time: 64.36s, d_loss: 1.3414 g_loss: 0.0283 (mse: 0.
0100, psnr: 26.8222, accuracy: 0.5469)
Epoch [10/20]   15 time: 64.29s, d_loss: 1.3055 g_loss: 0.0310 (mse: 0.
0161, psnr: 27.2233, accuracy: 0.5156)
Epoch [10/20]   16 time: 64.36s, d_loss: 1.3273 g_loss: 0.0258 (mse: 0.
0093, psnr: 27.7094, accuracy: 0.5312)
Epoch [10/20]   17 time: 64.66s, d_loss: 1.3381 g_loss: 0.0256 (mse: 0.
0089, psnr: 27.8292, accuracy: 0.5312)
Epoch [10/20]   18 time: 64.34s, d_loss: 1.3415 g_loss: 0.0234 (mse: 0.
0086, psnr: 27.9341, accuracy: 0.5469)
Epoch [10/20]   19 time: 64.94s, d_loss: 1.3417 g_loss: 0.0253 (mse: 0.
0096, psnr: 26.7523, accuracy: 0.5156)
Epoch [10/20]   20 time: 65.13s, d_loss: 1.3320 g_loss: 0.0296 (mse: 0.
0112, psnr: 26.6995, accuracy: 0.5156)
Epoch [10/20]   21 time: 65.26s, d_loss: 1.3765 g_loss: 0.0295 (mse: 0.
0113, psnr: 26.9857, accuracy: 0.5469)
Epoch [10/20]   22 time: 64.60s, d_loss: 1.3671 g_loss: 0.0279 (mse: 0.
0105, psnr: 26.5507, accuracy: 0.5469)
Epoch [10/20]   23 time: 64.45s, d_loss: 1.3321 g_loss: 0.0278 (mse: 0.
0116, psnr: 26.8291, accuracy: 0.5156)
Epoch [10/20]   24 time: 64.98s, d_loss: 1.3672 g_loss: 0.0286 (mse: 0.
0116, psnr: 26.6396, accuracy: 0.5000)
Epoch [10/20]   25 time: 64.58s, d_loss: 1.3418 g_loss: 0.0285 (mse: 0.
0109, psnr: 26.8302, accuracy: 0.5000)
Epoch [10/20]   26 time: 64.49s, d_loss: 1.3530 g_loss: 0.0288 (mse: 0.
0113, psnr: 26.7211, accuracy: 0.5000)
Epoch [10/20]   27 time: 64.41s, d_loss: 1.3091 g_loss: 0.0247 (mse: 0.
0103, psnr: 27.3460, accuracy: 0.5469)
Epoch [10/20]   28 time: 64.93s, d_loss: 1.3535 g_loss: 0.0256 (mse: 0.
0096, psnr: 27.0020, accuracy: 0.5156)
Epoch [10/20]   29 time: 65.24s, d_loss: 1.2934 g_loss: 0.0272 (mse: 0.
0104, psnr: 27.0814, accuracy: 0.5625)
Epoch [10/20]   30 time: 64.26s, d_loss: 1.3639 g_loss: 0.0260 (mse: 0.
0111, psnr: 26.8689, accuracy: 0.5000)
Epoch [10/20]   31 time: 64.48s, d_loss: 1.3452 g_loss: 0.0290 (mse: 0.
0120, psnr: 27.3053, accuracy: 0.5156)
Epoch [10/20]   32 time: 65.09s, d_loss: 1.3712 g_loss: 0.0234 (mse: 0.
0088, psnr: 27.8684, accuracy: 0.5469)
Epoch [10/20]   33 time: 65.10s, d_loss: 1.3839 g_loss: 0.0302 (mse: 0.
0130, psnr: 25.3277, accuracy: 0.5000)
Epoch [10/20]   34 time: 64.79s, d_loss: 1.3223 g_loss: 0.0238 (mse: 0.
0079, psnr: 27.7340, accuracy: 0.5000)
Epoch [10/20]   35 time: 64.42s, d_loss: 1.3216 g_loss: 0.0269 (mse: 0.
0097, psnr: 27.3963, accuracy: 0.5938)
Epoch [10/20]   36 time: 65.21s, d_loss: 1.3248 g_loss: 0.0239 (mse: 0.
0095, psnr: 27.5330, accuracy: 0.5000)
Epoch [10/20]   37 time: 64.32s, d_loss: 1.3136 g_loss: 0.0261 (mse: 0.
0097, psnr: 27.0846, accuracy: 0.5625)
Epoch [10/20]   38 time: 64.82s, d_loss: 1.3114 g_loss: 0.0266 (mse: 0.
0093, psnr: 27.3539, accuracy: 0.5312)
Epoch [10/20]   39 time: 64.41s, d_loss: 1.2910 g_loss: 0.0272 (mse: 0.
0098, psnr: 27.6035, accuracy: 0.5000)
Epoch [10/20]   40 time: 64.48s, d_loss: 1.3072 g_loss: 0.0245 (mse: 0.
0094, psnr: 27.7881, accuracy: 0.5156)
Epoch [10/20]   41 time: 63.70s, d_loss: 1.3184 g_loss: 0.0280 (mse: 0.
```

```
0104, psnr: 27.5995, accuracy: 0.5312)
Epoch [10/20]   42 time: 64.34s, d_loss: 1.3635 g_loss: 0.0228 (mse: 0.
0078, psnr: 27.9422, accuracy: 0.5000)
Epoch [10/20]   43 time: 64.29s, d_loss: 1.2999 g_loss: 0.0238 (mse: 0.
0076, psnr: 28.4730, accuracy: 0.5000)
Epoch [10/20]   44 time: 63.47s, d_loss: 1.2927 g_loss: 0.0239 (mse: 0.
0087, psnr: 27.6466, accuracy: 0.5469)
Epoch [10/20]   45 time: 64.75s, d_loss: 1.2786 g_loss: 0.0312 (mse: 0.
0116, psnr: 26.4760, accuracy: 0.5625)
Epoch [10/20]   46 time: 56.18s, d_loss: 1.2897 g_loss: 0.0270 (mse: 0.
0094, psnr: 27.1154, accuracy: 0.5893)
[*] Epoch: [10/20] time: 3042.33s, d_loss: 1.3377 g_loss: 0.0270 (mse:
 0.010264, psnr: 27.1691, accuracy: 0.5252)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [11/20]    0 time: 65.11s, d_loss: 1.3399 g_loss: 0.0207 (mse: 0.
0073, psnr: 28.3014, accuracy: 0.5000)
Epoch [11/20]    1 time: 64.34s, d_loss: 1.3120 g_loss: 0.0216 (mse: 0.
0084, psnr: 27.7638, accuracy: 0.5156)
Epoch [11/20]    2 time: 64.46s, d_loss: 1.3254 g_loss: 0.0300 (mse: 0.
0114, psnr: 26.2109, accuracy: 0.5469)
Epoch [11/20]    3 time: 64.32s, d_loss: 1.3732 g_loss: 0.0256 (mse: 0.
0094, psnr: 27.4769, accuracy: 0.5000)
Epoch [11/20]    4 time: 64.26s, d_loss: 1.2462 g_loss: 0.0276 (mse: 0.
0111, psnr: 27.3174, accuracy: 0.6719)
Epoch [11/20]    5 time: 64.47s, d_loss: 1.2679 g_loss: 0.0280 (mse: 0.
0110, psnr: 26.6593, accuracy: 0.6562)
Epoch [11/20]    6 time: 64.71s, d_loss: 1.2826 g_loss: 0.0314 (mse: 0.
0125, psnr: 26.8668, accuracy: 0.6562)
Epoch [11/20]    7 time: 64.85s, d_loss: 1.2821 g_loss: 0.0272 (mse: 0.
0097, psnr: 26.9271, accuracy: 0.5625)
Epoch [11/20]    8 time: 64.76s, d_loss: 1.3124 g_loss: 0.0195 (mse: 0.
0077, psnr: 28.5480, accuracy: 0.5938)
Epoch [11/20]    9 time: 64.70s, d_loss: 1.2565 g_loss: 0.0285 (mse: 0.
0103, psnr: 27.1698, accuracy: 0.6562)
Epoch [11/20]   10 time: 65.02s, d_loss: 1.3091 g_loss: 0.0242 (mse: 0.
0100, psnr: 27.3752, accuracy: 0.5625)
Epoch [11/20]   11 time: 64.39s, d_loss: 1.3555 g_loss: 0.0280 (mse: 0.
0133, psnr: 27.2089, accuracy: 0.5000)
Epoch [11/20]   12 time: 64.89s, d_loss: 1.2853 g_loss: 0.0302 (mse: 0.
0130, psnr: 26.5004, accuracy: 0.5156)
Epoch [11/20]   13 time: 64.74s, d_loss: 1.2687 g_loss: 0.0291 (mse: 0.
0113, psnr: 26.4967, accuracy: 0.5625)
Epoch [11/20]   14 time: 64.91s, d_loss: 1.2990 g_loss: 0.0305 (mse: 0.
0106, psnr: 26.6007, accuracy: 0.5625)
Epoch [11/20]   15 time: 64.52s, d_loss: 1.2573 g_loss: 0.0327 (mse: 0.
0128, psnr: 26.5160, accuracy: 0.5469)
Epoch [11/20]   16 time: 64.77s, d_loss: 1.2189 g_loss: 0.0313 (mse: 0.
0145, psnr: 26.1923, accuracy: 0.5469)
Epoch [11/20]   17 time: 64.48s, d_loss: 1.2171 g_loss: 0.0330 (mse: 0.
0125, psnr: 25.8643, accuracy: 0.5625)
Epoch [11/20]   18 time: 64.58s, d_loss: 1.2608 g_loss: 0.0287 (mse: 0.
0115, psnr: 26.5100, accuracy: 0.5000)
Epoch [11/20]   19 time: 64.56s, d_loss: 1.2751 g_loss: 0.0262 (mse: 0.
```

```
0097, psnr: 27.1107, accuracy: 0.5000)
Epoch [11/20]   20 time: 64.63s, d_loss: 1.2940 g_loss: 0.0248 (mse: 0.
0089, psnr: 27.5003, accuracy: 0.5781)
Epoch [11/20]   21 time: 64.29s, d_loss: 1.2683 g_loss: 0.0289 (mse: 0.
0100, psnr: 27.3305, accuracy: 0.5469)
Epoch [11/20]   22 time: 64.73s, d_loss: 1.3130 g_loss: 0.0223 (mse: 0.
0077, psnr: 27.5877, accuracy: 0.5156)
Epoch [11/20]   23 time: 64.74s, d_loss: 1.2579 g_loss: 0.0220 (mse: 0.
0081, psnr: 28.2738, accuracy: 0.6406)
Epoch [11/20]   24 time: 64.38s, d_loss: 1.2208 g_loss: 0.0288 (mse: 0.
0140, psnr: 26.4316, accuracy: 0.7031)
Epoch [11/20]   25 time: 64.86s, d_loss: 1.3031 g_loss: 0.0213 (mse: 0.
0069, psnr: 28.3193, accuracy: 0.5469)
Epoch [11/20]   26 time: 64.20s, d_loss: 1.2410 g_loss: 0.0300 (mse: 0.
0109, psnr: 26.5830, accuracy: 0.6094)
Epoch [11/20]   27 time: 63.80s, d_loss: 1.2731 g_loss: 0.0269 (mse: 0.
0113, psnr: 26.7325, accuracy: 0.5781)
Epoch [11/20]   28 time: 64.42s, d_loss: 1.2643 g_loss: 0.0247 (mse: 0.
0090, psnr: 27.8815, accuracy: 0.5156)
Epoch [11/20]   29 time: 65.10s, d_loss: 1.2646 g_loss: 0.0245 (mse: 0.
0086, psnr: 27.5867, accuracy: 0.5312)
Epoch [11/20]   30 time: 65.49s, d_loss: 1.4168 g_loss: 0.0262 (mse: 0.
0097, psnr: 26.7751, accuracy: 0.5000)
Epoch [11/20]   31 time: 63.25s, d_loss: 1.3047 g_loss: 0.0290 (mse: 0.
0149, psnr: 27.9117, accuracy: 0.5000)
Epoch [11/20]   32 time: 64.29s, d_loss: 1.2912 g_loss: 0.0241 (mse: 0.
0087, psnr: 27.8384, accuracy: 0.5156)
Epoch [11/20]   33 time: 63.75s, d_loss: 1.2918 g_loss: 0.0281 (mse: 0.
0106, psnr: 27.3029, accuracy: 0.5000)
Epoch [11/20]   34 time: 64.30s, d_loss: 1.2512 g_loss: 0.0338 (mse: 0.
0158, psnr: 26.3709, accuracy: 0.7031)
Epoch [11/20]   35 time: 64.20s, d_loss: 1.2936 g_loss: 0.0254 (mse: 0.
0110, psnr: 27.3846, accuracy: 0.5000)
Epoch [11/20]   36 time: 64.20s, d_loss: 1.1504 g_loss: 0.0323 (mse: 0.
0121, psnr: 26.4342, accuracy: 0.6406)
Epoch [11/20]   37 time: 64.07s, d_loss: 1.3074 g_loss: 0.0232 (mse: 0.
0089, psnr: 28.2320, accuracy: 0.5000)
Epoch [11/20]   38 time: 64.23s, d_loss: 1.2907 g_loss: 0.0335 (mse: 0.
0148, psnr: 26.9893, accuracy: 0.6719)
Epoch [11/20]   39 time: 64.26s, d_loss: 1.2353 g_loss: 0.0278 (mse: 0.
0100, psnr: 27.4360, accuracy: 0.6094)
Epoch [11/20]   40 time: 64.33s, d_loss: 1.2898 g_loss: 0.0225 (mse: 0.
0078, psnr: 28.0034, accuracy: 0.5625)
Epoch [11/20]   41 time: 63.99s, d_loss: 1.2398 g_loss: 0.0275 (mse: 0.
0122, psnr: 26.6246, accuracy: 0.6250)
Epoch [11/20]   42 time: 64.29s, d_loss: 1.1891 g_loss: 0.0261 (mse: 0.
0100, psnr: 27.7028, accuracy: 0.7188)
Epoch [11/20]   43 time: 64.00s, d_loss: 1.2811 g_loss: 0.0216 (mse: 0.
0075, psnr: 27.9676, accuracy: 0.5156)
Epoch [11/20]   44 time: 64.37s, d_loss: 1.2565 g_loss: 0.0255 (mse: 0.
0091, psnr: 27.5810, accuracy: 0.5000)
Epoch [11/20]   45 time: 64.66s, d_loss: 1.2239 g_loss: 0.0230 (mse: 0.
0085, psnr: 28.2001, accuracy: 0.5469)
Epoch [11/20]   46 time: 56.51s, d_loss: 1.3056 g_loss: 0.0248 (mse: 0.
0080, psnr: 27.9751, accuracy: 0.5179)
[*] Epoch: [11/20] time: 3022.21s, d_loss: 1.2780 g_loss: 0.0269 (mse:
 0.010484, psnr: 27.2462, accuracy: 0.5662)
```

```
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [12/20]    0 time: 65.10s, d_loss: 1.2710 g_loss: 0.0224 (mse: 0.
0081, psnr: 28.1073, accuracy: 0.5156)
Epoch [12/20]    1 time: 64.37s, d_loss: 1.2310 g_loss: 0.0221 (mse: 0.
0071, psnr: 28.5782, accuracy: 0.5469)
Epoch [12/20]    2 time: 64.44s, d_loss: 1.2855 g_loss: 0.0255 (mse: 0.
0096, psnr: 26.8701, accuracy: 0.5312)
Epoch [12/20]    3 time: 64.23s, d_loss: 1.2685 g_loss: 0.0244 (mse: 0.
0109, psnr: 27.0018, accuracy: 0.7031)
Epoch [12/20]    4 time: 64.35s, d_loss: 1.2362 g_loss: 0.0306 (mse: 0.
0116, psnr: 26.4198, accuracy: 0.6406)
Epoch [12/20]    5 time: 64.44s, d_loss: 1.2562 g_loss: 0.0276 (mse: 0.
0107, psnr: 27.0087, accuracy: 0.7812)
Epoch [12/20]    6 time: 64.52s, d_loss: 1.2242 g_loss: 0.0305 (mse: 0.
0114, psnr: 27.2553, accuracy: 0.7656)
Epoch [12/20]    7 time: 64.28s, d_loss: 1.2708 g_loss: 0.0262 (mse: 0.
0098, psnr: 27.4852, accuracy: 0.5156)
Epoch [12/20]    8 time: 64.38s, d_loss: 1.1939 g_loss: 0.0271 (mse: 0.
0099, psnr: 27.3203, accuracy: 0.6250)
Epoch [12/20]    9 time: 64.60s, d_loss: 1.1652 g_loss: 0.0297 (mse: 0.
0101, psnr: 27.2922, accuracy: 0.5312)
Epoch [12/20]   10 time: 64.25s, d_loss: 1.1849 g_loss: 0.0269 (mse: 0.
0109, psnr: 26.8105, accuracy: 0.5312)
Epoch [12/20]   11 time: 64.27s, d_loss: 1.2737 g_loss: 0.0258 (mse: 0.
0080, psnr: 28.6446, accuracy: 0.5156)
Epoch [12/20]   12 time: 64.45s, d_loss: 1.2782 g_loss: 0.0213 (mse: 0.
0078, psnr: 28.5390, accuracy: 0.5000)
Epoch [12/20]   13 time: 64.35s, d_loss: 1.1927 g_loss: 0.0294 (mse: 0.
0144, psnr: 26.4686, accuracy: 0.7188)
Epoch [12/20]   14 time: 64.20s, d_loss: 1.2410 g_loss: 0.0235 (mse: 0.
0079, psnr: 28.0221, accuracy: 0.5625)
Epoch [12/20]   15 time: 64.41s, d_loss: 1.1951 g_loss: 0.0232 (mse: 0.
0091, psnr: 28.5941, accuracy: 0.5312)
Epoch [12/20]   16 time: 64.24s, d_loss: 1.5097 g_loss: 0.0275 (mse: 0.
0113, psnr: 26.9074, accuracy: 0.5625)
Epoch [12/20]   17 time: 64.43s, d_loss: 1.1844 g_loss: 0.0271 (mse: 0.
0104, psnr: 26.9613, accuracy: 0.8125)
Epoch [12/20]   18 time: 64.23s, d_loss: 1.2378 g_loss: 0.0252 (mse: 0.
0094, psnr: 27.8312, accuracy: 0.6875)
Epoch [12/20]   19 time: 64.56s, d_loss: 1.1960 g_loss: 0.0279 (mse: 0.
0095, psnr: 27.2347, accuracy: 0.5312)
Epoch [12/20]   20 time: 64.19s, d_loss: 1.3037 g_loss: 0.0244 (mse: 0.
0082, psnr: 28.1155, accuracy: 0.5156)
Epoch [12/20]   21 time: 64.41s, d_loss: 1.2950 g_loss: 0.0307 (mse: 0.
0149, psnr: 26.8521, accuracy: 0.5000)
Epoch [12/20]   22 time: 64.41s, d_loss: 1.1525 g_loss: 0.0251 (mse: 0.
0087, psnr: 27.7182, accuracy: 0.5938)
Epoch [12/20]   23 time: 64.12s, d_loss: 1.2290 g_loss: 0.0256 (mse: 0.
0100, psnr: 27.3857, accuracy: 0.5312)
Epoch [12/20]   24 time: 64.47s, d_loss: 1.1936 g_loss: 0.0282 (mse: 0.
0098, psnr: 27.4689, accuracy: 0.5000)
Epoch [12/20]   25 time: 64.11s, d_loss: 1.2147 g_loss: 0.0255 (mse: 0.
0100, psnr: 27.2298, accuracy: 0.5625)
```

```
Epoch [12/20]   26 time: 64.71s, d_loss: 1.3029 g_loss: 0.0235 (mse: 0.
0088, psnr: 27.3577, accuracy: 0.5000)
Epoch [12/20]   27 time: 64.33s, d_loss: 1.1721 g_loss: 0.0309 (mse: 0.
0125, psnr: 26.3421, accuracy: 0.6094)
Epoch [12/20]   28 time: 64.50s, d_loss: 1.2688 g_loss: 0.0255 (mse: 0.
0099, psnr: 27.6981, accuracy: 0.7188)
Epoch [12/20]   29 time: 65.61s, d_loss: 1.2532 g_loss: 0.0255 (mse: 0.
0098, psnr: 27.2120, accuracy: 0.7812)
Epoch [12/20]   30 time: 68.45s, d_loss: 1.2733 g_loss: 0.0258 (mse: 0.
0108, psnr: 27.4146, accuracy: 0.6406)
Epoch [12/20]   31 time: 67.50s, d_loss: 1.2971 g_loss: 0.0223 (mse: 0.
0084, psnr: 28.1568, accuracy: 0.6875)
Epoch [12/20]   32 time: 64.43s, d_loss: 1.1299 g_loss: 0.0263 (mse: 0.
0099, psnr: 26.9434, accuracy: 0.6875)
Epoch [12/20]   33 time: 64.41s, d_loss: 1.2517 g_loss: 0.0244 (mse: 0.
0087, psnr: 28.1683, accuracy: 0.5000)
Epoch [12/20]   34 time: 64.61s, d_loss: 1.1140 g_loss: 0.0308 (mse: 0.
0104, psnr: 26.7907, accuracy: 0.5156)
Epoch [12/20]   35 time: 64.29s, d_loss: 1.1704 g_loss: 0.0270 (mse: 0.
0101, psnr: 27.1765, accuracy: 0.5469)
Epoch [12/20]   36 time: 64.68s, d_loss: 1.1894 g_loss: 0.0242 (mse: 0.
0087, psnr: 28.3326, accuracy: 0.6406)
Epoch [12/20]   37 time: 64.61s, d_loss: 1.0793 g_loss: 0.0328 (mse: 0.
0129, psnr: 25.9373, accuracy: 0.6562)
Epoch [12/20]   38 time: 65.14s, d_loss: 1.1875 g_loss: 0.0253 (mse: 0.
0104, psnr: 27.2567, accuracy: 0.6094)
Epoch [12/20]   39 time: 65.26s, d_loss: 1.0760 g_loss: 0.0304 (mse: 0.
0132, psnr: 26.6478, accuracy: 0.6250)
Epoch [12/20]   40 time: 65.05s, d_loss: 1.1060 g_loss: 0.0222 (mse: 0.
0075, psnr: 28.1822, accuracy: 0.7656)
Epoch [12/20]   41 time: 64.72s, d_loss: 1.0894 g_loss: 0.0295 (mse: 0.
0144, psnr: 27.1991, accuracy: 0.7812)
Epoch [12/20]   42 time: 66.52s, d_loss: 1.1424 g_loss: 0.0278 (mse: 0.
0103, psnr: 27.3797, accuracy: 0.8594)
Epoch [12/20]   43 time: 67.07s, d_loss: 1.0608 g_loss: 0.0223 (mse: 0.
0083, psnr: 27.3811, accuracy: 0.5469)
Epoch [12/20]   44 time: 67.00s, d_loss: 1.0635 g_loss: 0.0286 (mse: 0.
0104, psnr: 26.4939, accuracy: 0.5156)
Epoch [12/20]   45 time: 67.35s, d_loss: 1.1492 g_loss: 0.0273 (mse: 0.
0106, psnr: 27.1708, accuracy: 0.5000)
Epoch [12/20]   46 time: 59.49s, d_loss: 1.1336 g_loss: 0.0254 (mse: 0.
0096, psnr: 27.5057, accuracy: 0.5893)
[*] Epoch: [12/20] time: 3043.54s, d_loss: 1.2084 g_loss: 0.0264 (mse:
 0.010109, psnr: 27.3802, accuracy: 0.6083)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [13/20]    0 time: 66.27s, d_loss: 1.0696 g_loss: 0.0269 (mse: 0.
0097, psnr: 26.9522, accuracy: 0.5938)
Epoch [13/20]    1 time: 69.28s, d_loss: 1.0504 g_loss: 0.0263 (mse: 0.
0104, psnr: 27.4316, accuracy: 0.7812)
Epoch [13/20]    2 time: 64.72s, d_loss: 1.0700 g_loss: 0.0231 (mse: 0.
0094, psnr: 27.1972, accuracy: 0.5938)
Epoch [13/20]    3 time: 64.93s, d_loss: 1.1635 g_loss: 0.0247 (mse: 0.
0090, psnr: 27.3854, accuracy: 0.7812)
```

```
Epoch [13/20]    4 time: 64.67s, d_loss: 1.3134 g_loss: 0.0325 (mse: 0.
0128, psnr: 26.0310, accuracy: 0.7031)
Epoch [13/20]    5 time: 66.55s, d_loss: 1.0465 g_loss: 0.0268 (mse: 0.
0106, psnr: 27.4724, accuracy: 0.6719)
Epoch [13/20]    6 time: 64.96s, d_loss: 1.3556 g_loss: 0.0218 (mse: 0.
0074, psnr: 28.3332, accuracy: 0.5000)
Epoch [13/20]    7 time: 64.79s, d_loss: 1.3438 g_loss: 0.0233 (mse: 0.
0086, psnr: 28.1477, accuracy: 0.5156)
Epoch [13/20]    8 time: 64.23s, d_loss: 1.1479 g_loss: 0.0245 (mse: 0.
0093, psnr: 27.3102, accuracy: 0.5312)
Epoch [13/20]    9 time: 64.48s, d_loss: 1.1858 g_loss: 0.0236 (mse: 0.
0085, psnr: 27.9661, accuracy: 0.6719)
Epoch [13/20]   10 time: 65.01s, d_loss: 1.1370 g_loss: 0.0271 (mse: 0.
0090, psnr: 27.3236, accuracy: 0.7969)
Epoch [13/20]   11 time: 64.79s, d_loss: 1.0450 g_loss: 0.1480 (mse: 0.
1292, psnr: 26.0791, accuracy: 0.8594)
Epoch [13/20]   12 time: 64.78s, d_loss: 1.1181 g_loss: 0.0309 (mse: 0.
0104, psnr: 26.5984, accuracy: 0.8281)
Epoch [13/20]   13 time: 64.38s, d_loss: 1.2703 g_loss: 0.0274 (mse: 0.
0101, psnr: 26.5910, accuracy: 0.5000)
Epoch [13/20]   14 time: 64.94s, d_loss: 1.3702 g_loss: 0.0297 (mse: 0.
0109, psnr: 26.3350, accuracy: 0.5000)
Epoch [13/20]   15 time: 64.91s, d_loss: 1.5035 g_loss: 0.0246 (mse: 0.
0095, psnr: 27.3437, accuracy: 0.5000)
Epoch [13/20]   16 time: 64.79s, d_loss: 0.9428 g_loss: 0.0336 (mse: 0.
0127, psnr: 26.1153, accuracy: 0.7500)
Epoch [13/20]   17 time: 64.54s, d_loss: 1.1584 g_loss: 0.0270 (mse: 0.
0109, psnr: 26.9855, accuracy: 0.5000)
Epoch [13/20]   18 time: 64.84s, d_loss: 0.9716 g_loss: 0.0285 (mse: 0.
0129, psnr: 26.5719, accuracy: 0.9531)
Epoch [13/20]   19 time: 64.92s, d_loss: 1.1184 g_loss: 0.0283 (mse: 0.
0108, psnr: 26.4571, accuracy: 0.8906)
Epoch [13/20]   20 time: 66.06s, d_loss: 1.1671 g_loss: 0.0211 (mse: 0.
0072, psnr: 28.5122, accuracy: 0.6875)
Epoch [13/20]   21 time: 64.65s, d_loss: 0.9636 g_loss: 0.0251 (mse: 0.
0111, psnr: 27.1569, accuracy: 0.8281)
Epoch [13/20]   22 time: 65.16s, d_loss: 1.1639 g_loss: 0.0237 (mse: 0.
0081, psnr: 27.8106, accuracy: 0.5625)
Epoch [13/20]   23 time: 65.47s, d_loss: 1.1854 g_loss: 0.0325 (mse: 0.
0158, psnr: 26.2780, accuracy: 0.8438)
Epoch [13/20]   24 time: 64.85s, d_loss: 1.1483 g_loss: 0.0224 (mse: 0.
0084, psnr: 27.9802, accuracy: 0.5625)
Epoch [13/20]   25 time: 64.85s, d_loss: 1.0420 g_loss: 0.0238 (mse: 0.
0086, psnr: 27.7332, accuracy: 0.6094)
Epoch [13/20]   26 time: 64.94s, d_loss: 1.2591 g_loss: 0.0212 (mse: 0.
0069, psnr: 28.7190, accuracy: 0.5312)
Epoch [13/20]   27 time: 64.98s, d_loss: 1.0908 g_loss: 0.0257 (mse: 0.
0106, psnr: 27.6195, accuracy: 0.5312)
Epoch [13/20]   28 time: 67.15s, d_loss: 1.0527 g_loss: 0.0266 (mse: 0.
0104, psnr: 27.3486, accuracy: 0.7500)
Epoch [13/20]   29 time: 65.54s, d_loss: 0.9670 g_loss: 0.0256 (mse: 0.
0109, psnr: 27.3590, accuracy: 0.8438)
Epoch [13/20]   30 time: 65.08s, d_loss: 0.8227 g_loss: 0.0339 (mse: 0.
0133, psnr: 25.5252, accuracy: 0.9844)
Epoch [13/20]   31 time: 64.31s, d_loss: 1.1496 g_loss: 0.0245 (mse: 0.
0086, psnr: 27.7253, accuracy: 0.5938)
Epoch [13/20]   32 time: 64.55s, d_loss: 0.9639 g_loss: 0.0323 (mse: 0.
```

```
0126, psnr: 26.3690, accuracy: 0.9219)
Epoch [13/20]   33 time: 64.59s, d_loss: 1.4307 g_loss: 0.0202 (mse: 0.
0068, psnr: 28.6127, accuracy: 0.5000)
Epoch [13/20]   34 time: 64.32s, d_loss: 1.1982 g_loss: 0.0252 (mse: 0.
0087, psnr: 28.0530, accuracy: 0.5156)
Epoch [13/20]   35 time: 67.20s, d_loss: 0.9723 g_loss: 0.0317 (mse: 0.
0153, psnr: 27.1477, accuracy: 0.7500)
Epoch [13/20]   36 time: 66.34s, d_loss: 1.0178 g_loss: 0.0231 (mse: 0.
0083, psnr: 27.9049, accuracy: 0.9062)
Epoch [13/20]   37 time: 66.00s, d_loss: 1.1152 g_loss: 0.0236 (mse: 0.
0101, psnr: 27.8142, accuracy: 0.7812)
Epoch [13/20]   38 time: 65.75s, d_loss: 1.1043 g_loss: 0.0261 (mse: 0.
0092, psnr: 27.1085, accuracy: 0.7656)
Epoch [13/20]   39 time: 64.50s, d_loss: 1.0664 g_loss: 0.0270 (mse: 0.
0096, psnr: 27.4618, accuracy: 0.7344)
Epoch [13/20]   40 time: 64.64s, d_loss: 0.9932 g_loss: 0.0284 (mse: 0.
0107, psnr: 27.1688, accuracy: 0.8125)
Epoch [13/20]   41 time: 64.99s, d_loss: 0.9615 g_loss: 0.0322 (mse: 0.
0152, psnr: 27.5928, accuracy: 0.6250)
Epoch [13/20]   42 time: 64.74s, d_loss: 1.2072 g_loss: 0.0246 (mse: 0.
0088, psnr: 27.5181, accuracy: 0.5312)
Epoch [13/20]   43 time: 66.94s, d_loss: 1.0714 g_loss: 0.0231 (mse: 0.
0093, psnr: 28.0945, accuracy: 0.5625)
Epoch [13/20]   44 time: 69.93s, d_loss: 0.8366 g_loss: 0.0291 (mse: 0.
0106, psnr: 27.1331, accuracy: 0.7344)
Epoch [13/20]   45 time: 65.44s, d_loss: 1.0757 g_loss: 0.0261 (mse: 0.
0091, psnr: 27.5822, accuracy: 0.5625)
Epoch [13/20]   46 time: 56.80s, d_loss: 1.5266 g_loss: 0.0344 (mse: 0.
0143, psnr: 25.2085, accuracy: 0.5357)
[*] Epoch: [13/20] time: 3062.58s, d_loss: 1.1263 g_loss: 0.0292 (mse:
 0.012778, psnr: 27.2588, accuracy: 0.6806)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [14/20]    0 time: 65.72s, d_loss: 0.9203 g_loss: 0.0279 (mse: 0.
0101, psnr: 27.4335, accuracy: 0.9844)
Epoch [14/20]    1 time: 64.89s, d_loss: 0.9281 g_loss: 0.0312 (mse: 0.
0146, psnr: 27.2264, accuracy: 0.6719)
Epoch [14/20]    2 time: 65.29s, d_loss: 1.2276 g_loss: 0.0217 (mse: 0.
0080, psnr: 27.8711, accuracy: 0.5000)
Epoch [14/20]    3 time: 65.04s, d_loss: 1.6059 g_loss: 0.0326 (mse: 0.
0134, psnr: 25.8004, accuracy: 0.5000)
Epoch [14/20]    4 time: 64.89s, d_loss: 1.1696 g_loss: 0.0259 (mse: 0.
0108, psnr: 26.8771, accuracy: 0.5156)
Epoch [14/20]    5 time: 64.85s, d_loss: 0.7791 g_loss: 0.0322 (mse: 0.
0148, psnr: 25.9331, accuracy: 0.8125)
Epoch [14/20]    6 time: 64.84s, d_loss: 0.9715 g_loss: 0.0249 (mse: 0.
0092, psnr: 27.5675, accuracy: 0.6406)
Epoch [14/20]    7 time: 64.52s, d_loss: 1.3087 g_loss: 0.0248 (mse: 0.
0095, psnr: 28.0345, accuracy: 0.5469)
Epoch [14/20]    8 time: 64.50s, d_loss: 1.3355 g_loss: 0.0265 (mse: 0.
0101, psnr: 27.2485, accuracy: 0.6094)
Epoch [14/20]    9 time: 64.47s, d_loss: 1.1269 g_loss: 0.0270 (mse: 0.
0087, psnr: 27.8533, accuracy: 0.8438)
Epoch [14/20]   10 time: 64.75s, d_loss: 1.3770 g_loss: 0.0262 (mse: 0.
```

```
0102, psnr: 26.9576, accuracy: 0.6094)
Epoch [14/20]   11 time: 64.47s, d_loss: 1.1512 g_loss: 0.0237 (mse: 0.
0083, psnr: 27.7467, accuracy: 0.7969)
Epoch [14/20]   12 time: 64.26s, d_loss: 1.1153 g_loss: 0.0295 (mse: 0.
0109, psnr: 26.4069, accuracy: 0.5000)
Epoch [14/20]   13 time: 64.48s, d_loss: 1.4920 g_loss: 0.0233 (mse: 0.
0083, psnr: 27.6783, accuracy: 0.5000)
Epoch [14/20]   14 time: 64.91s, d_loss: 1.2071 g_loss: 0.0238 (mse: 0.
0084, psnr: 27.4300, accuracy: 0.5000)
Epoch [14/20]   15 time: 64.65s, d_loss: 0.7455 g_loss: 0.0269 (mse: 0.
0102, psnr: 27.2132, accuracy: 0.9688)
Epoch [14/20]   16 time: 64.76s, d_loss: 1.1411 g_loss: 0.0266 (mse: 0.
0108, psnr: 26.9667, accuracy: 0.5938)
Epoch [14/20]   17 time: 65.58s, d_loss: 1.1257 g_loss: 0.0288 (mse: 0.
0117, psnr: 26.9034, accuracy: 0.8438)
Epoch [14/20]   18 time: 67.06s, d_loss: 1.1941 g_loss: 0.0281 (mse: 0.
0102, psnr: 26.7379, accuracy: 0.7656)
Epoch [14/20]   19 time: 64.49s, d_loss: 1.0294 g_loss: 0.0227 (mse: 0.
0084, psnr: 27.6700, accuracy: 0.9531)
Epoch [14/20]   20 time: 65.65s, d_loss: 0.9583 g_loss: 0.0302 (mse: 0.
0142, psnr: 27.2722, accuracy: 0.8594)
Epoch [14/20]   21 time: 65.30s, d_loss: 0.8652 g_loss: 0.0293 (mse: 0.
0130, psnr: 27.1113, accuracy: 0.6406)
Epoch [14/20]   22 time: 65.93s, d_loss: 1.4307 g_loss: 0.0246 (mse: 0.
0079, psnr: 28.1969, accuracy: 0.5000)
Epoch [14/20]   23 time: 65.23s, d_loss: 1.2841 g_loss: 0.0245 (mse: 0.
0082, psnr: 27.9285, accuracy: 0.5000)
Epoch [14/20]   24 time: 65.00s, d_loss: 0.9158 g_loss: 0.0231 (mse: 0.
0081, psnr: 27.8524, accuracy: 0.7812)
Epoch [14/20]   25 time: 64.61s, d_loss: 1.1004 g_loss: 0.0295 (mse: 0.
0115, psnr: 26.4817, accuracy: 0.8438)
Epoch [14/20]   26 time: 65.62s, d_loss: 0.8429 g_loss: 0.0330 (mse: 0.
0135, psnr: 25.3543, accuracy: 0.9375)
Epoch [14/20]   27 time: 64.67s, d_loss: 1.1811 g_loss: 0.0245 (mse: 0.
0091, psnr: 27.1660, accuracy: 0.5000)
Epoch [14/20]   28 time: 64.64s, d_loss: 0.8296 g_loss: 0.0273 (mse: 0.
0103, psnr: 27.2632, accuracy: 0.9844)
Epoch [14/20]   29 time: 66.22s, d_loss: 1.0198 g_loss: 0.0272 (mse: 0.
0091, psnr: 27.6498, accuracy: 0.6562)
Epoch [14/20]   30 time: 65.31s, d_loss: 1.3398 g_loss: 0.0227 (mse: 0.
0092, psnr: 27.2593, accuracy: 0.6250)
Epoch [14/20]   31 time: 64.63s, d_loss: 0.8153 g_loss: 0.0272 (mse: 0.
0091, psnr: 27.4420, accuracy: 0.9688)
Epoch [14/20]   32 time: 64.66s, d_loss: 1.0508 g_loss: 0.0282 (mse: 0.
0106, psnr: 27.0387, accuracy: 0.6094)
Epoch [14/20]   33 time: 65.63s, d_loss: 1.4241 g_loss: 0.0267 (mse: 0.
0097, psnr: 27.2558, accuracy: 0.5000)
Epoch [14/20]   34 time: 64.20s, d_loss: 1.0571 g_loss: 0.0269 (mse: 0.
0099, psnr: 27.0745, accuracy: 0.5469)
Epoch [14/20]   35 time: 64.90s, d_loss: 1.1893 g_loss: 0.0290 (mse: 0.
0113, psnr: 26.5327, accuracy: 0.5156)
Epoch [14/20]   36 time: 63.32s, d_loss: 1.1062 g_loss: 0.0248 (mse: 0.
0087, psnr: 27.5407, accuracy: 0.5312)
Epoch [14/20]   37 time: 64.82s, d_loss: 1.0285 g_loss: 0.0330 (mse: 0.
0133, psnr: 26.1495, accuracy: 0.8281)
Epoch [14/20]   38 time: 64.92s, d_loss: 1.1841 g_loss: 0.0267 (mse: 0.
0103, psnr: 26.5083, accuracy: 0.7656)
```

```
Epoch [14/20]   39 time: 64.82s, d_loss: 1.1677 g_loss: 0.0271 (mse: 0.
0124, psnr: 26.7755, accuracy: 0.8281)
Epoch [14/20]   40 time: 64.33s, d_loss: 0.7318 g_loss: 0.0316 (mse: 0.
0106, psnr: 26.8902, accuracy: 0.9531)
Epoch [14/20]   41 time: 64.35s, d_loss: 1.2126 g_loss: 0.0250 (mse: 0.
0094, psnr: 27.3983, accuracy: 0.5000)
Epoch [14/20]   42 time: 64.64s, d_loss: 0.9216 g_loss: 0.0302 (mse: 0.
0106, psnr: 27.3061, accuracy: 0.5156)
Epoch [14/20]   43 time: 64.54s, d_loss: 0.9690 g_loss: 0.0254 (mse: 0.
0093, psnr: 27.2358, accuracy: 0.6094)
Epoch [14/20]   44 time: 64.68s, d_loss: 0.9110 g_loss: 0.0291 (mse: 0.
0125, psnr: 26.8633, accuracy: 0.6562)
Epoch [14/20]   45 time: 63.90s, d_loss: 1.1658 g_loss: 0.0216 (mse: 0.
0077, psnr: 28.0923, accuracy: 0.5469)
Epoch [14/20]   46 time: 56.63s, d_loss: 1.3843 g_loss: 0.0178 (mse: 0.
0067, psnr: 28.7552, accuracy: 0.6607)
[*] Epoch: [14/20] time: 3041.54s, d_loss: 1.1072 g_loss: 0.0268 (mse:
 0.010272, psnr: 27.1904, accuracy: 0.6813)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [15/20]    0 time: 65.92s, d_loss: 1.1234 g_loss: 0.0239 (mse: 0.
0099, psnr: 27.9759, accuracy: 0.8281)
Epoch [15/20]    1 time: 65.17s, d_loss: 1.3697 g_loss: 0.0247 (mse: 0.
0091, psnr: 27.6929, accuracy: 0.5938)
Epoch [15/20]    2 time: 64.70s, d_loss: 1.0510 g_loss: 0.0280 (mse: 0.
0112, psnr: 26.6269, accuracy: 0.8750)
Epoch [15/20]    3 time: 64.72s, d_loss: 0.9443 g_loss: 0.0252 (mse: 0.
0092, psnr: 27.3504, accuracy: 0.7812)
Epoch [15/20]    4 time: 64.51s, d_loss: 1.1851 g_loss: 0.0256 (mse: 0.
0090, psnr: 27.6181, accuracy: 0.5000)
Epoch [15/20]    5 time: 64.08s, d_loss: 1.1410 g_loss: 0.0330 (mse: 0.
0118, psnr: 26.1952, accuracy: 0.5156)
Epoch [15/20]    6 time: 64.51s, d_loss: 1.7573 g_loss: 0.0282 (mse: 0.
0099, psnr: 26.7551, accuracy: 0.5000)
Epoch [15/20]    7 time: 64.43s, d_loss: 1.1299 g_loss: 0.0262 (mse: 0.
0101, psnr: 27.1205, accuracy: 0.5312)
Epoch [15/20]    8 time: 64.39s, d_loss: 1.1824 g_loss: 0.0313 (mse: 0.
0145, psnr: 26.5166, accuracy: 0.5469)
Epoch [15/20]    9 time: 64.43s, d_loss: 1.0054 g_loss: 0.0251 (mse: 0.
0100, psnr: 26.8799, accuracy: 0.7969)
Epoch [15/20]   10 time: 64.50s, d_loss: 1.2779 g_loss: 0.0244 (mse: 0.
0088, psnr: 27.7084, accuracy: 0.7500)
Epoch [15/20]   11 time: 64.60s, d_loss: 1.1186 g_loss: 0.0245 (mse: 0.
0103, psnr: 27.3972, accuracy: 0.8281)
Epoch [15/20]   12 time: 64.65s, d_loss: 1.2890 g_loss: 0.0286 (mse: 0.
0110, psnr: 26.6583, accuracy: 0.6875)
Epoch [15/20]   13 time: 64.82s, d_loss: 0.9999 g_loss: 0.0235 (mse: 0.
0080, psnr: 28.6053, accuracy: 0.8594)
Epoch [15/20]   14 time: 64.49s, d_loss: 1.1358 g_loss: 0.0250 (mse: 0.
0085, psnr: 27.6624, accuracy: 0.5625)
Epoch [15/20]   15 time: 64.93s, d_loss: 0.9940 g_loss: 0.0215 (mse: 0.
0071, psnr: 28.8204, accuracy: 0.5469)
Epoch [15/20]   16 time: 64.54s, d_loss: 0.8488 g_loss: 0.0220 (mse: 0.
0080, psnr: 28.1446, accuracy: 0.8594)
```

```
Epoch [15/20]   17 time: 64.69s, d_loss: 0.8340 g_loss: 0.0250 (mse: 0.
0088, psnr: 27.7237, accuracy: 1.0000)
Epoch [15/20]   18 time: 63.76s, d_loss: 1.3084 g_loss: 0.0204 (mse: 0.
0071, psnr: 28.3697, accuracy: 0.5000)
Epoch [15/20]   19 time: 64.45s, d_loss: 0.9554 g_loss: 0.0229 (mse: 0.
0086, psnr: 27.9288, accuracy: 0.7656)
Epoch [15/20]   20 time: 64.75s, d_loss: 0.9545 g_loss: 0.0324 (mse: 0.
0110, psnr: 26.5627, accuracy: 0.9688)
Epoch [15/20]   21 time: 64.80s, d_loss: 0.9246 g_loss: 0.0289 (mse: 0.
0093, psnr: 27.3278, accuracy: 1.0000)
Epoch [15/20]   22 time: 64.58s, d_loss: 1.3290 g_loss: 0.0251 (mse: 0.
0092, psnr: 27.3238, accuracy: 0.5000)
Epoch [15/20]   23 time: 64.50s, d_loss: 1.0338 g_loss: 0.0314 (mse: 0.
0130, psnr: 26.2184, accuracy: 0.5469)
Epoch [15/20]   24 time: 64.64s, d_loss: 1.3382 g_loss: 0.0262 (mse: 0.
0098, psnr: 27.1048, accuracy: 0.5938)
Epoch [15/20]   25 time: 64.22s, d_loss: 1.0514 g_loss: 0.0226 (mse: 0.
0081, psnr: 28.0994, accuracy: 0.5781)
Epoch [15/20]   26 time: 64.64s, d_loss: 0.7834 g_loss: 0.0263 (mse: 0.
0093, psnr: 27.4989, accuracy: 0.9375)
Epoch [15/20]   27 time: 64.37s, d_loss: 0.9481 g_loss: 0.0216 (mse: 0.
0079, psnr: 28.0589, accuracy: 0.5938)
Epoch [15/20]   28 time: 64.48s, d_loss: 0.9684 g_loss: 0.0279 (mse: 0.
0110, psnr: 26.2451, accuracy: 0.8750)
Epoch [15/20]   29 time: 64.72s, d_loss: 1.5324 g_loss: 0.0245 (mse: 0.
0088, psnr: 27.9919, accuracy: 0.5000)
Epoch [15/20]   30 time: 63.97s, d_loss: 0.9944 g_loss: 0.0268 (mse: 0.
0093, psnr: 27.8673, accuracy: 0.5469)
Epoch [15/20]   31 time: 64.71s, d_loss: 0.7802 g_loss: 0.0271 (mse: 0.
0100, psnr: 27.6173, accuracy: 1.0000)
Epoch [15/20]   32 time: 64.62s, d_loss: 0.7934 g_loss: 0.0314 (mse: 0.
0116, psnr: 26.8777, accuracy: 0.9219)
Epoch [15/20]   33 time: 64.63s, d_loss: 0.9231 g_loss: 0.0277 (mse: 0.
0105, psnr: 26.5963, accuracy: 0.8906)
Epoch [15/20]   34 time: 64.24s, d_loss: 0.9899 g_loss: 0.0261 (mse: 0.
0095, psnr: 27.4971, accuracy: 0.7188)
Epoch [15/20]   35 time: 64.85s, d_loss: 0.8202 g_loss: 0.0275 (mse: 0.
0091, psnr: 27.2338, accuracy: 0.8438)
Epoch [15/20]   36 time: 64.54s, d_loss: 1.2116 g_loss: 0.0263 (mse: 0.
0122, psnr: 27.1757, accuracy: 0.5469)
Epoch [15/20]   37 time: 64.73s, d_loss: 1.1566 g_loss: 0.0241 (mse: 0.
0091, psnr: 27.5707, accuracy: 0.7969)
Epoch [15/20]   38 time: 66.98s, d_loss: 0.9381 g_loss: 0.0410 (mse: 0.
0216, psnr: 26.5363, accuracy: 0.8438)
Epoch [15/20]   39 time: 1271.33s, d_loss: 1.1763 g_loss: 0.0222 (mse:
 0.0077, psnr: 28.5372, accuracy: 0.5000)
Epoch [15/20]   40 time: 75.74s, d_loss: 0.9772 g_loss: 0.0272 (mse: 0.
0096, psnr: 27.5081, accuracy: 0.6094)
Epoch [15/20]   41 time: 67.13s, d_loss: 0.8741 g_loss: 0.0276 (mse: 0.
0105, psnr: 27.2044, accuracy: 0.9844)
Epoch [15/20]   42 time: 64.81s, d_loss: 0.8636 g_loss: 0.0304 (mse: 0.
0116, psnr: 27.0772, accuracy: 0.9219)
Epoch [15/20]   43 time: 64.37s, d_loss: 1.0671 g_loss: 0.0297 (mse: 0.
0130, psnr: 25.9263, accuracy: 0.5000)
Epoch [15/20]   44 time: 64.28s, d_loss: 0.9126 g_loss: 0.0305 (mse: 0.
0112, psnr: 26.3225, accuracy: 0.8281)
Epoch [15/20]   45 time: 64.73s, d_loss: 1.0114 g_loss: 0.0271 (mse: 0.
```

```
0101, psnr: 27.0617, accuracy: 0.5312)
Epoch [15/20]   46 time: 56.14s, d_loss: 1.0334 g_loss: 0.0285 (mse: 0.
0106, psnr: 26.2554, accuracy: 0.8929)
[*] Epoch: [15/20] time: 4249.80s, d_loss: 1.0646 g_loss: 0.0267 (mse:
 0.010117, psnr: 27.2989, accuracy: 0.7191)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [16/20]    0 time: 64.40s, d_loss: 1.8561 g_loss: 0.0201 (mse: 0.
0072, psnr: 28.3475, accuracy: 0.5000)
Epoch [16/20]    1 time: 64.33s, d_loss: 0.9626 g_loss: 0.0299 (mse: 0.
0106, psnr: 27.2416, accuracy: 0.9219)
Epoch [16/20]    2 time: 64.82s, d_loss: 0.9590 g_loss: 0.0394 (mse: 0.
0197, psnr: 26.6032, accuracy: 0.9219)
Epoch [16/20]    3 time: 65.21s, d_loss: 1.2102 g_loss: 0.0282 (mse: 0.
0124, psnr: 27.2562, accuracy: 0.5469)
Epoch [16/20]    4 time: 65.37s, d_loss: 1.3159 g_loss: 0.0322 (mse: 0.
0161, psnr: 26.6820, accuracy: 0.5156)
Epoch [16/20]    5 time: 65.45s, d_loss: 0.9095 g_loss: 0.0316 (mse: 0.
0111, psnr: 26.7906, accuracy: 0.6406)
Epoch [16/20]    6 time: 65.40s, d_loss: 1.3357 g_loss: 0.0208 (mse: 0.
0069, psnr: 28.6398, accuracy: 0.5156)
Epoch [16/20]    7 time: 65.79s, d_loss: 0.7913 g_loss: 0.0257 (mse: 0.
0104, psnr: 27.1195, accuracy: 0.9688)
Epoch [16/20]    8 time: 64.41s, d_loss: 1.0388 g_loss: 0.0229 (mse: 0.
0085, psnr: 27.7953, accuracy: 0.7969)
Epoch [16/20]    9 time: 64.03s, d_loss: 1.1497 g_loss: 0.0228 (mse: 0.
0088, psnr: 27.4054, accuracy: 0.7812)
Epoch [16/20]   10 time: 63.76s, d_loss: 1.1112 g_loss: 0.0263 (mse: 0.
0096, psnr: 27.3669, accuracy: 0.8125)
Epoch [16/20]   11 time: 64.59s, d_loss: 0.9874 g_loss: 0.0279 (mse: 0.
0105, psnr: 27.0566, accuracy: 0.5938)
Epoch [16/20]   12 time: 64.91s, d_loss: 0.7199 g_loss: 0.0256 (mse: 0.
0091, psnr: 27.5519, accuracy: 0.9531)
Epoch [16/20]   13 time: 64.07s, d_loss: 0.8219 g_loss: 0.0297 (mse: 0.
0108, psnr: 26.8112, accuracy: 0.6562)
Epoch [16/20]   14 time: 64.97s, d_loss: 0.7085 g_loss: 0.0237 (mse: 0.
0079, psnr: 28.3098, accuracy: 0.8125)
Epoch [16/20]   15 time: 64.89s, d_loss: 0.7492 g_loss: 0.0378 (mse: 0.
0185, psnr: 26.7197, accuracy: 0.9688)
Epoch [16/20]   16 time: 65.17s, d_loss: 0.8957 g_loss: 0.0257 (mse: 0.
0089, psnr: 27.6498, accuracy: 0.7031)
Epoch [16/20]   17 time: 65.44s, d_loss: 0.8958 g_loss: 0.0213 (mse: 0.
0080, psnr: 27.9294, accuracy: 0.7188)
Epoch [16/20]   18 time: 64.71s, d_loss: 0.6249 g_loss: 0.0315 (mse: 0.
0123, psnr: 26.4569, accuracy: 1.0000)
Epoch [16/20]   19 time: 64.69s, d_loss: 0.8504 g_loss: 0.0264 (mse: 0.
0093, psnr: 27.6687, accuracy: 0.7812)
Epoch [16/20]   20 time: 64.87s, d_loss: 0.9237 g_loss: 0.0253 (mse: 0.
0094, psnr: 27.6688, accuracy: 0.6406)
Epoch [16/20]   21 time: 65.06s, d_loss: 0.7906 g_loss: 0.0292 (mse: 0.
0120, psnr: 26.3889, accuracy: 1.0000)
Epoch [16/20]   22 time: 65.44s, d_loss: 0.6212 g_loss: 0.0250 (mse: 0.
0095, psnr: 27.7179, accuracy: 0.8906)
Epoch [16/20]   23 time: 64.01s, d_loss: 0.6759 g_loss: 0.0301 (mse: 0.
```

```
0104, psnr: 26.9996, accuracy: 1.0000)
Epoch [16/20]   24 time: 65.25s, d_loss: 0.7143 g_loss: 0.0297 (mse: 0.
0118, psnr: 26.6892, accuracy: 0.7031)
Epoch [16/20]   25 time: 63.98s, d_loss: 0.7712 g_loss: 0.0295 (mse: 0.
0106, psnr: 26.7932, accuracy: 0.6719)
Epoch [16/20]   26 time: 65.47s, d_loss: 0.6254 g_loss: 0.0289 (mse: 0.
0104, psnr: 26.9827, accuracy: 1.0000)
Epoch [16/20]   27 time: 64.28s, d_loss: 0.8578 g_loss: 0.0278 (mse: 0.
0095, psnr: 27.6589, accuracy: 0.6250)
Epoch [16/20]   28 time: 64.37s, d_loss: 0.7671 g_loss: 0.0275 (mse: 0.
0100, psnr: 27.5592, accuracy: 0.8906)
Epoch [16/20]   29 time: 64.21s, d_loss: 1.0849 g_loss: 0.0277 (mse: 0.
0102, psnr: 27.0518, accuracy: 0.7344)
Epoch [16/20]   30 time: 65.47s, d_loss: 1.1445 g_loss: 0.0293 (mse: 0.
0118, psnr: 26.3223, accuracy: 0.5156)
Epoch [16/20]   31 time: 65.12s, d_loss: 1.3193 g_loss: 0.0251 (mse: 0.
0083, psnr: 27.5924, accuracy: 0.5000)
Epoch [16/20]   32 time: 64.69s, d_loss: 0.8913 g_loss: 0.0249 (mse: 0.
0084, psnr: 28.0840, accuracy: 0.9531)
Epoch [16/20]   33 time: 64.87s, d_loss: 0.7420 g_loss: 0.0288 (mse: 0.
0105, psnr: 27.0893, accuracy: 0.9844)
Epoch [16/20]   34 time: 64.36s, d_loss: 0.8529 g_loss: 0.0255 (mse: 0.
0097, psnr: 26.7267, accuracy: 0.9062)
Epoch [16/20]   35 time: 65.11s, d_loss: 0.5668 g_loss: 0.0286 (mse: 0.
0112, psnr: 27.2861, accuracy: 0.9688)
Epoch [16/20]   36 time: 66.23s, d_loss: 1.1115 g_loss: 0.0221 (mse: 0.
0081, psnr: 28.0835, accuracy: 0.6094)
Epoch [16/20]   37 time: 64.63s, d_loss: 0.9848 g_loss: 0.0295 (mse: 0.
0111, psnr: 27.4223, accuracy: 0.5156)
Epoch [16/20]   38 time: 64.88s, d_loss: 0.8517 g_loss: 0.0261 (mse: 0.
0096, psnr: 28.0092, accuracy: 0.7188)
Epoch [16/20]   39 time: 68.67s, d_loss: 0.8321 g_loss: 0.0326 (mse: 0.
0143, psnr: 26.7218, accuracy: 0.9531)
Epoch [16/20]   40 time: 68.37s, d_loss: 1.1233 g_loss: 0.0255 (mse: 0.
0099, psnr: 27.1150, accuracy: 0.5000)
Epoch [16/20]   41 time: 69.94s, d_loss: 1.4128 g_loss: 0.0261 (mse: 0.
0109, psnr: 26.1219, accuracy: 0.6719)
Epoch [16/20]   42 time: 69.32s, d_loss: 1.0283 g_loss: 0.0219 (mse: 0.
0080, psnr: 27.7859, accuracy: 0.9062)
Epoch [16/20]   43 time: 67.64s, d_loss: 1.5373 g_loss: 0.0282 (mse: 0.
0115, psnr: 25.9778, accuracy: 0.5312)
Epoch [16/20]   44 time: 68.46s, d_loss: 1.0609 g_loss: 0.0263 (mse: 0.
0084, psnr: 27.6584, accuracy: 0.5156)
Epoch [16/20]   45 time: 68.91s, d_loss: 0.7541 g_loss: 0.0286 (mse: 0.
0095, psnr: 26.7958, accuracy: 0.7188)
Epoch [16/20]   46 time: 58.86s, d_loss: 0.9791 g_loss: 0.0281 (mse: 0.
0125, psnr: 26.6042, accuracy: 0.5000)
[*] Epoch: [16/20] time: 3068.91s, d_loss: 0.9557 g_loss: 0.0274 (mse:
 0.010512, psnr: 27.2406, accuracy: 0.7497)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [17/20]    0 time: 70.78s, d_loss: 1.2023 g_loss: 0.0233 (mse: 0.
0093, psnr: 27.8014, accuracy: 0.8125)
Epoch [17/20]    1 time: 71.62s, d_loss: 0.7764 g_loss: 0.0299 (mse: 0.
```

```
0125, psnr: 26.5435, accuracy: 0.9688)
Epoch [17/20]    2 time: 71.28s, d_loss: 1.6237 g_loss: 0.0261 (mse: 0.
0093, psnr: 27.3945, accuracy: 0.5156)
Epoch [17/20]    3 time: 69.32s, d_loss: 0.6274 g_loss: 0.0293 (mse: 0.
0108, psnr: 26.9180, accuracy: 0.7812)
Epoch [17/20]    4 time: 68.12s, d_loss: 1.5001 g_loss: 0.0393 (mse: 0.
0217, psnr: 26.3135, accuracy: 0.5000)
Epoch [17/20]    5 time: 68.10s, d_loss: 0.7425 g_loss: 0.0310 (mse: 0.
0110, psnr: 26.5751, accuracy: 0.6250)
Epoch [17/20]    6 time: 70.07s, d_loss: 1.5330 g_loss: 0.0234 (mse: 0.
0088, psnr: 27.5454, accuracy: 0.5000)
Epoch [17/20]    7 time: 71.82s, d_loss: 1.0035 g_loss: 0.0238 (mse: 0.
0087, psnr: 28.1420, accuracy: 0.9375)
Epoch [17/20]    8 time: 69.90s, d_loss: 0.9785 g_loss: 0.0255 (mse: 0.
0091, psnr: 27.2927, accuracy: 0.8906)
Epoch [17/20]    9 time: 68.75s, d_loss: 1.2271 g_loss: 0.0272 (mse: 0.
0093, psnr: 27.5689, accuracy: 0.6875)
Epoch [17/20]   10 time: 70.19s, d_loss: 1.0149 g_loss: 0.0337 (mse: 0.
0154, psnr: 25.8689, accuracy: 0.7656)
Epoch [17/20]   11 time: 67.79s, d_loss: 0.9765 g_loss: 0.0259 (mse: 0.
0085, psnr: 28.1230, accuracy: 0.5156)
Epoch [17/20]   12 time: 68.42s, d_loss: 1.5163 g_loss: 0.0298 (mse: 0.
0139, psnr: 27.2838, accuracy: 0.5000)
Epoch [17/20]   13 time: 66.39s, d_loss: 1.3874 g_loss: 0.0260 (mse: 0.
0087, psnr: 27.7199, accuracy: 0.5000)
Epoch [17/20]   14 time: 68.39s, d_loss: 0.9968 g_loss: 0.0288 (mse: 0.
0117, psnr: 26.6175, accuracy: 0.6562)
Epoch [17/20]   15 time: 68.18s, d_loss: 1.1119 g_loss: 0.0294 (mse: 0.
0105, psnr: 26.3286, accuracy: 0.5312)
Epoch [17/20]   16 time: 70.14s, d_loss: 1.1670 g_loss: 0.0290 (mse: 0.
0102, psnr: 27.4056, accuracy: 0.7031)
Epoch [17/20]   17 time: 68.23s, d_loss: 1.1785 g_loss: 0.0261 (mse: 0.
0107, psnr: 27.3561, accuracy: 0.7031)
Epoch [17/20]   18 time: 64.84s, d_loss: 1.0807 g_loss: 0.0263 (mse: 0.
0096, psnr: 27.4830, accuracy: 0.8281)
Epoch [17/20]   19 time: 65.31s, d_loss: 0.9560 g_loss: 0.0262 (mse: 0.
0096, psnr: 27.2130, accuracy: 0.7344)
Epoch [17/20]   20 time: 68.26s, d_loss: 1.6072 g_loss: 0.0252 (mse: 0.
0120, psnr: 26.8569, accuracy: 0.5000)
Epoch [17/20]   21 time: 66.03s, d_loss: 0.9673 g_loss: 0.0263 (mse: 0.
0103, psnr: 27.5108, accuracy: 0.8281)
Epoch [17/20]   22 time: 67.48s, d_loss: 1.1201 g_loss: 0.0289 (mse: 0.
0097, psnr: 27.0801, accuracy: 0.5625)
Epoch [17/20]   23 time: 67.97s, d_loss: 1.4025 g_loss: 0.0290 (mse: 0.
0107, psnr: 26.3348, accuracy: 0.5156)
Epoch [17/20]   24 time: 70.33s, d_loss: 1.2351 g_loss: 0.0260 (mse: 0.
0121, psnr: 27.3917, accuracy: 0.5625)
Epoch [17/20]   25 time: 67.60s, d_loss: 1.1483 g_loss: 0.0244 (mse: 0.
0091, psnr: 27.1223, accuracy: 0.5938)
Epoch [17/20]   26 time: 68.14s, d_loss: 1.2260 g_loss: 0.0225 (mse: 0.
0073, psnr: 28.2934, accuracy: 0.7969)
Epoch [17/20]   27 time: 71.21s, d_loss: 1.3122 g_loss: 0.0289 (mse: 0.
0110, psnr: 26.7910, accuracy: 0.6719)
Epoch [17/20]   28 time: 73.35s, d_loss: 1.1308 g_loss: 0.0206 (mse: 0.
0073, psnr: 28.5398, accuracy: 0.7188)
Epoch [17/20]   29 time: 69.40s, d_loss: 0.8943 g_loss: 0.0257 (mse: 0.
0089, psnr: 27.6919, accuracy: 0.8281)
```

```
Epoch [17/20]   30 time: 66.06s, d_loss: 0.8617 g_loss: 0.0251 (mse: 0.
0090, psnr: 27.6462, accuracy: 0.9219)
Epoch [17/20]   31 time: 66.46s, d_loss: 0.9191 g_loss: 0.0260 (mse: 0.
0104, psnr: 26.8832, accuracy: 0.6562)
Epoch [17/20]   32 time: 65.86s, d_loss: 0.6138 g_loss: 0.0276 (mse: 0.
0101, psnr: 27.3894, accuracy: 1.0000)
Epoch [17/20]   33 time: 67.60s, d_loss: 0.9450 g_loss: 0.0312 (mse: 0.
0160, psnr: 26.1860, accuracy: 0.8281)
Epoch [17/20]   34 time: 73.69s, d_loss: 1.0575 g_loss: 0.0279 (mse: 0.
0101, psnr: 27.1447, accuracy: 0.7656)
Epoch [17/20]   35 time: 66.37s, d_loss: 0.6357 g_loss: 0.0247 (mse: 0.
0082, psnr: 27.9185, accuracy: 0.9688)
Epoch [17/20]   36 time: 65.61s, d_loss: 0.4642 g_loss: 0.0321 (mse: 0.
0126, psnr: 25.9539, accuracy: 0.9219)
Epoch [17/20]   37 time: 67.42s, d_loss: 2.1407 g_loss: 0.0264 (mse: 0.
0090, psnr: 27.6886, accuracy: 0.5000)
Epoch [17/20]   38 time: 68.93s, d_loss: 0.7992 g_loss: 0.0340 (mse: 0.
0119, psnr: 25.7837, accuracy: 0.5781)
Epoch [17/20]   39 time: 68.20s, d_loss: 1.7970 g_loss: 0.0309 (mse: 0.
0132, psnr: 27.2643, accuracy: 0.5000)
Epoch [17/20]   40 time: 68.74s, d_loss: 1.1205 g_loss: 0.0268 (mse: 0.
0089, psnr: 27.4182, accuracy: 0.5625)
Epoch [17/20]   41 time: 65.95s, d_loss: 0.7514 g_loss: 0.0246 (mse: 0.
0090, psnr: 27.6054, accuracy: 0.9062)
Epoch [17/20]   42 time: 66.16s, d_loss: 1.0436 g_loss: 0.0236 (mse: 0.
0096, psnr: 27.6766, accuracy: 0.8281)
Epoch [17/20]   43 time: 66.42s, d_loss: 1.5931 g_loss: 0.0278 (mse: 0.
0101, psnr: 27.1178, accuracy: 0.5312)
Epoch [17/20]   44 time: 65.17s, d_loss: 1.1199 g_loss: 0.0196 (mse: 0.
0071, psnr: 28.5213, accuracy: 0.8750)
Epoch [17/20]   45 time: 65.30s, d_loss: 0.8992 g_loss: 0.0274 (mse: 0.
0094, psnr: 27.4259, accuracy: 0.9531)
Epoch [17/20]   46 time: 58.49s, d_loss: 0.5751 g_loss: 0.0276 (mse: 0.
0105, psnr: 27.2438, accuracy: 1.0000)
[*] Epoch: [17/20] time: 3199.83s, d_loss: 1.1060 g_loss: 0.0273 (mse:
 0.010488, psnr: 27.2335, accuracy: 0.7134)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [18/20]    0 time: 67.07s, d_loss: 1.1310 g_loss: 0.0272 (mse: 0.
0099, psnr: 27.0507, accuracy: 0.5000)
Epoch [18/20]    1 time: 71.10s, d_loss: 1.3359 g_loss: 0.0240 (mse: 0.
0076, psnr: 28.2835, accuracy: 0.5000)
Epoch [18/20]    2 time: 70.69s, d_loss: 1.6057 g_loss: 0.0237 (mse: 0.
0078, psnr: 27.8038, accuracy: 0.5000)
Epoch [18/20]    3 time: 68.27s, d_loss: 1.2404 g_loss: 0.0247 (mse: 0.
0086, psnr: 28.2999, accuracy: 0.5000)
Epoch [18/20]    4 time: 69.36s, d_loss: 1.9627 g_loss: 0.0213 (mse: 0.
0072, psnr: 28.7068, accuracy: 0.5000)
Epoch [18/20]    5 time: 64.91s, d_loss: 0.8546 g_loss: 0.0282 (mse: 0.
0111, psnr: 26.8100, accuracy: 1.0000)
Epoch [18/20]    6 time: 68.59s, d_loss: 1.0719 g_loss: 0.0255 (mse: 0.
0094, psnr: 27.6553, accuracy: 0.7500)
Epoch [18/20]    7 time: 66.98s, d_loss: 1.7080 g_loss: 0.0273 (mse: 0.
0110, psnr: 26.5899, accuracy: 0.5156)
```

```
Epoch [18/20]    8 time: 70.01s, d_loss: 1.1575 g_loss: 0.0277 (mse: 0.
0104, psnr: 27.1191, accuracy: 0.6875)
Epoch [18/20]    9 time: 68.25s, d_loss: 1.0637 g_loss: 0.0334 (mse: 0.
0161, psnr: 26.2795, accuracy: 0.7500)
Epoch [18/20]   10 time: 66.04s, d_loss: 0.8046 g_loss: 0.0259 (mse: 0.
0094, psnr: 27.3180, accuracy: 1.0000)
Epoch [18/20]   11 time: 66.09s, d_loss: 0.4512 g_loss: 0.0283 (mse: 0.
0115, psnr: 26.8260, accuracy: 1.0000)
Epoch [18/20]   12 time: 66.37s, d_loss: 1.7381 g_loss: 0.0266 (mse: 0.
0091, psnr: 27.5544, accuracy: 0.5000)
Epoch [18/20]   13 time: 65.15s, d_loss: 0.9601 g_loss: 0.0268 (mse: 0.
0103, psnr: 27.1214, accuracy: 0.5781)
Epoch [18/20]   14 time: 65.20s, d_loss: 1.0226 g_loss: 0.0273 (mse: 0.
0095, psnr: 27.1689, accuracy: 0.5625)
Epoch [18/20]   15 time: 65.94s, d_loss: 0.5187 g_loss: 0.0304 (mse: 0.
0135, psnr: 26.6193, accuracy: 1.0000)
Epoch [18/20]   16 time: 66.30s, d_loss: 0.7306 g_loss: 0.0264 (mse: 0.
0094, psnr: 27.8141, accuracy: 0.9688)
Epoch [18/20]   17 time: 66.09s, d_loss: 0.8157 g_loss: 0.0244 (mse: 0.
0093, psnr: 27.6182, accuracy: 0.9531)
Epoch [18/20]   18 time: 66.51s, d_loss: 0.9937 g_loss: 0.0267 (mse: 0.
0093, psnr: 27.2119, accuracy: 0.8750)
Epoch [18/20]   19 time: 65.66s, d_loss: 0.9000 g_loss: 0.0331 (mse: 0.
0114, psnr: 26.2791, accuracy: 0.9375)
Epoch [18/20]   20 time: 65.86s, d_loss: 0.7611 g_loss: 0.0285 (mse: 0.
0107, psnr: 26.5528, accuracy: 0.9688)
Epoch [18/20]   21 time: 65.35s, d_loss: 0.9107 g_loss: 0.0269 (mse: 0.
0086, psnr: 27.4934, accuracy: 0.5156)
Epoch [18/20]   22 time: 65.46s, d_loss: 0.7190 g_loss: 0.0304 (mse: 0.
0122, psnr: 27.7646, accuracy: 0.6875)
Epoch [18/20]   23 time: 66.73s, d_loss: 1.1863 g_loss: 0.0302 (mse: 0.
0100, psnr: 26.5125, accuracy: 0.5000)
Epoch [18/20]   24 time: 64.42s, d_loss: 0.9823 g_loss: 0.0262 (mse: 0.
0091, psnr: 27.4195, accuracy: 0.5312)
Epoch [18/20]   25 time: 65.49s, d_loss: 0.7089 g_loss: 0.0261 (mse: 0.
0091, psnr: 27.3026, accuracy: 0.8750)
Epoch [18/20]   26 time: 65.42s, d_loss: 0.7444 g_loss: 0.0299 (mse: 0.
0122, psnr: 26.1366, accuracy: 0.9844)
Epoch [18/20]   27 time: 64.73s, d_loss: 0.9207 g_loss: 0.0230 (mse: 0.
0087, psnr: 27.7450, accuracy: 0.8906)
Epoch [18/20]   28 time: 64.86s, d_loss: 0.9809 g_loss: 0.0288 (mse: 0.
0108, psnr: 27.0822, accuracy: 0.8594)
Epoch [18/20]   29 time: 65.33s, d_loss: 0.9120 g_loss: 0.0270 (mse: 0.
0094, psnr: 27.2554, accuracy: 0.9062)
Epoch [18/20]   30 time: 65.57s, d_loss: 0.5623 g_loss: 0.0328 (mse: 0.
0127, psnr: 26.1415, accuracy: 0.9844)
Epoch [18/20]   31 time: 64.57s, d_loss: 1.2304 g_loss: 0.0293 (mse: 0.
0107, psnr: 27.1589, accuracy: 0.5000)
Epoch [18/20]   32 time: 64.72s, d_loss: 1.3692 g_loss: 0.0285 (mse: 0.
0103, psnr: 27.2988, accuracy: 0.5000)
Epoch [18/20]   33 time: 65.45s, d_loss: 0.8558 g_loss: 0.0293 (mse: 0.
0100, psnr: 27.1838, accuracy: 0.5312)
Epoch [18/20]   34 time: 65.51s, d_loss: 1.3433 g_loss: 0.0240 (mse: 0.
0088, psnr: 27.3924, accuracy: 0.5000)
Epoch [18/20]   35 time: 64.73s, d_loss: 0.6413 g_loss: 0.0292 (mse: 0.
0109, psnr: 26.5802, accuracy: 1.0000)
Epoch [18/20]   36 time: 64.75s, d_loss: 1.4991 g_loss: 0.0191 (mse: 0.
```

```
0063, psnr: 28.8901, accuracy: 0.5000)
Epoch [18/20]   37 time: 65.01s, d_loss: 1.8508 g_loss: 0.0257 (mse: 0.
0104, psnr: 26.8560, accuracy: 0.5000)
Epoch [18/20]   38 time: 66.51s, d_loss: 1.0333 g_loss: 0.0242 (mse: 0.
0102, psnr: 27.1792, accuracy: 0.8438)
Epoch [18/20]   39 time: 64.92s, d_loss: 0.9736 g_loss: 0.0239 (mse: 0.
0085, psnr: 27.5163, accuracy: 0.9219)
Epoch [18/20]   40 time: 66.14s, d_loss: 0.7499 g_loss: 0.0226 (mse: 0.
0075, psnr: 28.3706, accuracy: 0.9844)
Epoch [18/20]   41 time: 71.76s, d_loss: 0.9610 g_loss: 0.0246 (mse: 0.
0093, psnr: 27.3269, accuracy: 0.5625)
Epoch [18/20]   42 time: 70.28s, d_loss: 0.8338 g_loss: 0.0288 (mse: 0.
0111, psnr: 27.4203, accuracy: 0.6562)
Epoch [18/20]   43 time: 70.10s, d_loss: 1.1058 g_loss: 0.0267 (mse: 0.
0093, psnr: 27.8851, accuracy: 0.5156)
Epoch [18/20]   44 time: 70.59s, d_loss: 0.8691 g_loss: 0.0262 (mse: 0.
0091, psnr: 27.5414, accuracy: 0.6875)
Epoch [18/20]   45 time: 69.53s, d_loss: 0.7913 g_loss: 0.0233 (mse: 0.
0090, psnr: 27.3312, accuracy: 0.8594)
Epoch [18/20]   46 time: 59.72s, d_loss: 0.8714 g_loss: 0.0252 (mse: 0.
0085, psnr: 27.6170, accuracy: 0.5536)
[*] Epoch: [18/20] time: 3128.11s, d_loss: 1.0305 g_loss: 0.0268 (mse:
 0.009897, psnr: 27.2997, accuracy: 0.7212)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
Epoch [19/20]    0 time: 71.90s, d_loss: 1.1868 g_loss: 0.0278 (mse: 0.
0102, psnr: 26.7908, accuracy: 0.6250)
Epoch [19/20]    1 time: 69.91s, d_loss: 0.7775 g_loss: 0.0210 (mse: 0.
0069, psnr: 28.5742, accuracy: 0.9844)
Epoch [19/20]    2 time: 71.57s, d_loss: 1.0788 g_loss: 0.0235 (mse: 0.
0081, psnr: 28.4127, accuracy: 0.8125)
Epoch [19/20]    3 time: 71.08s, d_loss: 0.8043 g_loss: 0.0245 (mse: 0.
0074, psnr: 27.9686, accuracy: 0.7500)
Epoch [19/20]    4 time: 70.25s, d_loss: 0.4752 g_loss: 0.0276 (mse: 0.
0098, psnr: 27.4567, accuracy: 0.9688)
Epoch [19/20]    5 time: 68.52s, d_loss: 0.7462 g_loss: 0.0258 (mse: 0.
0090, psnr: 27.1974, accuracy: 0.6719)
Epoch [19/20]    6 time: 70.26s, d_loss: 0.4787 g_loss: 0.0292 (mse: 0.
0099, psnr: 26.9069, accuracy: 0.9688)
Epoch [19/20]    7 time: 69.41s, d_loss: 0.7498 g_loss: 0.0252 (mse: 0.
0094, psnr: 27.2561, accuracy: 0.7188)
Epoch [19/20]    8 time: 69.14s, d_loss: 0.7351 g_loss: 0.0274 (mse: 0.
0093, psnr: 27.5488, accuracy: 1.0000)
Epoch [19/20]    9 time: 69.15s, d_loss: 0.8668 g_loss: 0.0280 (mse: 0.
0112, psnr: 26.8692, accuracy: 0.9531)
Epoch [19/20]   10 time: 71.73s, d_loss: 0.8628 g_loss: 0.0231 (mse: 0.
0093, psnr: 27.3770, accuracy: 0.9844)
Epoch [19/20]   11 time: 68.45s, d_loss: 0.8806 g_loss: 0.0256 (mse: 0.
0086, psnr: 27.3828, accuracy: 0.6562)
Epoch [19/20]   12 time: 69.79s, d_loss: 0.8146 g_loss: 0.0281 (mse: 0.
0103, psnr: 26.7596, accuracy: 0.7812)
Epoch [19/20]   13 time: 71.35s, d_loss: 0.4779 g_loss: 0.0282 (mse: 0.
0118, psnr: 26.3513, accuracy: 1.0000)
Epoch [19/20]   14 time: 71.44s, d_loss: 0.2645 g_loss: 0.0322 (mse: 0.
```

```
0107, psnr: 26.2940, accuracy: 1.0000)
Epoch [19/20]   15 time: 71.83s, d_loss: 0.2698 g_loss: 0.0301 (mse: 0.
0109, psnr: 27.0057, accuracy: 1.0000)
Epoch [19/20]   16 time: 71.62s, d_loss: 0.6768 g_loss: 0.0288 (mse: 0.
0098, psnr: 26.9932, accuracy: 0.7344)
Epoch [19/20]   17 time: 71.33s, d_loss: 1.0239 g_loss: 0.0256 (mse: 0.
0104, psnr: 27.7383, accuracy: 0.5156)
Epoch [19/20]   18 time: 71.59s, d_loss: 0.5268 g_loss: 0.0227 (mse: 0.
0077, psnr: 28.4934, accuracy: 0.9375)
Epoch [19/20]   19 time: 71.57s, d_loss: 1.0784 g_loss: 0.0287 (mse: 0.
0123, psnr: 26.8165, accuracy: 0.7500)
Epoch [19/20]   20 time: 70.96s, d_loss: 0.5261 g_loss: 0.0294 (mse: 0.
0097, psnr: 27.1874, accuracy: 1.0000)
Epoch [19/20]   21 time: 70.83s, d_loss: 0.7160 g_loss: 0.0275 (mse: 0.
0096, psnr: 26.9767, accuracy: 1.0000)
Epoch [19/20]   22 time: 72.14s, d_loss: 0.5386 g_loss: 0.0269 (mse: 0.
0093, psnr: 27.6382, accuracy: 0.9844)
Epoch [19/20]   23 time: 69.67s, d_loss: 1.6508 g_loss: 0.0297 (mse: 0.
0136, psnr: 26.7623, accuracy: 0.5000)
Epoch [19/20]   24 time: 68.87s, d_loss: 1.0725 g_loss: 0.0231 (mse: 0.
0082, psnr: 28.0832, accuracy: 0.5469)
Epoch [19/20]   25 time: 76.03s, d_loss: 1.9767 g_loss: 0.0252 (mse: 0.
0081, psnr: 27.3532, accuracy: 0.5000)
Epoch [19/20]   26 time: 71.19s, d_loss: 1.1881 g_loss: 0.0277 (mse: 0.
0098, psnr: 27.2244, accuracy: 0.5000)
Epoch [19/20]   27 time: 71.24s, d_loss: 0.9055 g_loss: 0.0221 (mse: 0.
0082, psnr: 28.1421, accuracy: 0.7812)
Epoch [19/20]   28 time: 76.21s, d_loss: 1.2324 g_loss: 0.0280 (mse: 0.
0113, psnr: 26.0476, accuracy: 0.5156)
Epoch [19/20]   29 time: 70.81s, d_loss: 1.0025 g_loss: 0.0298 (mse: 0.
0119, psnr: 26.8431, accuracy: 0.7656)
Epoch [19/20]   30 time: 71.03s, d_loss: 1.0085 g_loss: 0.0287 (mse: 0.
0111, psnr: 26.9831, accuracy: 0.7500)
Epoch [19/20]   31 time: 70.88s, d_loss: 1.2542 g_loss: 0.0311 (mse: 0.
0153, psnr: 26.4523, accuracy: 0.7031)
Epoch [19/20]   32 time: 70.74s, d_loss: 0.8481 g_loss: 0.0217 (mse: 0.
0081, psnr: 28.2012, accuracy: 0.9531)
Epoch [19/20]   33 time: 70.42s, d_loss: 1.0922 g_loss: 0.0262 (mse: 0.
0118, psnr: 27.2325, accuracy: 0.8906)
Epoch [19/20]   34 time: 71.12s, d_loss: 0.6804 g_loss: 0.0235 (mse: 0.
0080, psnr: 27.9943, accuracy: 0.9531)
Epoch [19/20]   35 time: 70.85s, d_loss: 0.5508 g_loss: 0.0330 (mse: 0.
0118, psnr: 26.3995, accuracy: 1.0000)
Epoch [19/20]   36 time: 70.96s, d_loss: 0.7591 g_loss: 0.0354 (mse: 0.
0134, psnr: 25.2357, accuracy: 0.6719)
Epoch [19/20]   37 time: 68.73s, d_loss: 0.5059 g_loss: 0.0324 (mse: 0.
0117, psnr: 26.5200, accuracy: 1.0000)
Epoch [19/20]   38 time: 66.75s, d_loss: 0.3676 g_loss: 0.0259 (mse: 0.
0108, psnr: 27.5342, accuracy: 1.0000)
Epoch [19/20]   39 time: 64.82s, d_loss: 0.6002 g_loss: 0.0275 (mse: 0.
0105, psnr: 26.6635, accuracy: 0.9844)
Epoch [19/20]   40 time: 67.52s, d_loss: 0.3871 g_loss: 0.0332 (mse: 0.
0119, psnr: 26.2647, accuracy: 1.0000)
Epoch [19/20]   41 time: 66.17s, d_loss: 0.3094 g_loss: 0.0261 (mse: 0.
0102, psnr: 27.2878, accuracy: 1.0000)
Epoch [19/20]   42 time: 64.97s, d_loss: 0.5105 g_loss: 0.0293 (mse: 0.
0111, psnr: 26.6210, accuracy: 0.8594)
```

```
Epoch [19/20]   43 time: 65.12s, d_loss: 0.4886 g_loss: 0.0283 (mse: 0.
0110, psnr: 26.3316, accuracy: 0.9844)
Epoch [19/20]   44 time: 68.52s, d_loss: 0.8805 g_loss: 0.0251 (mse: 0.
0089, psnr: 27.4106, accuracy: 0.5156)
Epoch [19/20]   45 time: 70.22s, d_loss: 0.4789 g_loss: 0.0267 (mse: 0.
0100, psnr: 27.0465, accuracy: 1.0000)
Epoch [19/20]   46 time: 60.27s, d_loss: 1.1567 g_loss: 0.0215 (mse: 0.
0072, psnr: 28.1440, accuracy: 0.8571)
[*] Epoch: [19/20] time: 3288.96s, d_loss: 0.7971 g_loss: 0.0272 (mse:
 0.010112, psnr: 27.1654, accuracy: 0.8304)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
 ** new learning rate: 0.000001 (for GAN)
Epoch [20/20]    0 time: 71.93s, d_loss: 1.0205 g_loss: 0.0248 (mse: 0.
0091, psnr: 27.8191, accuracy: 0.7656)
Epoch [20/20]    1 time: 73.08s, d_loss: 1.0497 g_loss: 0.0315 (mse: 0.
0153, psnr: 26.2655, accuracy: 0.7812)
Epoch [20/20]    2 time: 71.02s, d_loss: 0.9050 g_loss: 0.0269 (mse: 0.
0098, psnr: 27.4885, accuracy: 0.9219)
Epoch [20/20]    3 time: 71.77s, d_loss: 0.9178 g_loss: 0.0235 (mse: 0.
0093, psnr: 27.3399, accuracy: 0.8906)
Epoch [20/20]    4 time: 70.65s, d_loss: 0.7086 g_loss: 0.0269 (mse: 0.
0103, psnr: 26.9007, accuracy: 1.0000)
Epoch [20/20]    5 time: 70.84s, d_loss: 0.8367 g_loss: 0.0303 (mse: 0.
0109, psnr: 26.9341, accuracy: 0.8438)
Epoch [20/20]    6 time: 70.59s, d_loss: 0.6276 g_loss: 0.0261 (mse: 0.
0090, psnr: 27.8329, accuracy: 0.9375)
Epoch [20/20]    7 time: 68.07s, d_loss: 0.9123 g_loss: 0.0234 (mse: 0.
0079, psnr: 28.2162, accuracy: 0.5938)
Epoch [20/20]    8 time: 68.72s, d_loss: 0.5644 g_loss: 0.0268 (mse: 0.
0089, psnr: 27.4912, accuracy: 0.8750)
Epoch [20/20]    9 time: 73.07s, d_loss: 0.8814 g_loss: 0.0245 (mse: 0.
0099, psnr: 27.7254, accuracy: 0.5312)
Epoch [20/20]   10 time: 73.28s, d_loss: 1.3701 g_loss: 0.0312 (mse: 0.
0153, psnr: 27.1235, accuracy: 0.5000)
Epoch [20/20]   11 time: 67.26s, d_loss: 0.5376 g_loss: 0.0265 (mse: 0.
0088, psnr: 27.3364, accuracy: 1.0000)
Epoch [20/20]   12 time: 71.64s, d_loss: 0.3805 g_loss: 0.0249 (mse: 0.
0092, psnr: 27.2145, accuracy: 1.0000)
Epoch [20/20]   13 time: 70.16s, d_loss: 0.5897 g_loss: 0.0287 (mse: 0.
0113, psnr: 26.4957, accuracy: 1.0000)
Epoch [20/20]   14 time: 70.04s, d_loss: 0.5680 g_loss: 0.0265 (mse: 0.
0096, psnr: 27.4662, accuracy: 0.9531)
Epoch [20/20]   15 time: 70.56s, d_loss: 1.4677 g_loss: 0.0224 (mse: 0.
0072, psnr: 28.2334, accuracy: 0.5000)
Epoch [20/20]   16 time: 68.25s, d_loss: 0.8095 g_loss: 0.0284 (mse: 0.
0103, psnr: 26.7296, accuracy: 0.6719)
Epoch [20/20]   17 time: 64.84s, d_loss: 0.3686 g_loss: 0.0289 (mse: 0.
0096, psnr: 27.1037, accuracy: 0.9844)
Epoch [20/20]   18 time: 70.11s, d_loss: 0.9853 g_loss: 0.0266 (mse: 0.
0093, psnr: 27.7498, accuracy: 0.5781)
Epoch [20/20]   19 time: 68.92s, d_loss: 0.7119 g_loss: 0.0256 (mse: 0.
0091, psnr: 27.4129, accuracy: 0.6719)
Epoch [20/20]   20 time: 69.06s, d_loss: 0.4014 g_loss: 0.0308 (mse: 0.
```

```
0121, psnr: 26.4048, accuracy: 1.0000)
Epoch [20/20]   21 time: 67.85s, d_loss: 0.7251 g_loss: 0.0280 (mse: 0.
0101, psnr: 27.3528, accuracy: 1.0000)
Epoch [20/20]   22 time: 76.09s, d_loss: 0.7066 g_loss: 0.0297 (mse: 0.
0117, psnr: 26.2381, accuracy: 0.9844)
Epoch [20/20]   23 time: 70.86s, d_loss: 0.6808 g_loss: 0.0235 (mse: 0.
0098, psnr: 26.7901, accuracy: 0.8906)
Epoch [20/20]   24 time: 69.28s, d_loss: 0.4586 g_loss: 0.0281 (mse: 0.
0101, psnr: 27.4833, accuracy: 1.0000)
Epoch [20/20]   25 time: 68.05s, d_loss: 0.7406 g_loss: 0.0253 (mse: 0.
0094, psnr: 27.0285, accuracy: 0.7969)
Epoch [20/20]   26 time: 71.03s, d_loss: 0.6323 g_loss: 0.0239 (mse: 0.
0083, psnr: 28.0225, accuracy: 0.8125)
Epoch [20/20]   27 time: 73.65s, d_loss: 0.4490 g_loss: 0.0307 (mse: 0.
0104, psnr: 26.5882, accuracy: 0.9844)
Epoch [20/20]   28 time: 70.73s, d_loss: 0.5793 g_loss: 0.0279 (mse: 0.
0100, psnr: 27.5947, accuracy: 1.0000)
Epoch [20/20]   29 time: 74.93s, d_loss: 0.4149 g_loss: 0.0286 (mse: 0.
0105, psnr: 27.2461, accuracy: 1.0000)
Epoch [20/20]   30 time: 74.68s, d_loss: 0.2589 g_loss: 0.0253 (mse: 0.
0090, psnr: 27.4951, accuracy: 1.0000)
Epoch [20/20]   31 time: 73.17s, d_loss: 0.5018 g_loss: 0.0290 (mse: 0.
0102, psnr: 26.5353, accuracy: 1.0000)
Epoch [20/20]   32 time: 72.77s, d_loss: 0.4637 g_loss: 0.0314 (mse: 0.
0128, psnr: 27.4319, accuracy: 1.0000)
Epoch [20/20]   33 time: 74.07s, d_loss: 0.5587 g_loss: 0.0310 (mse: 0.
0109, psnr: 26.8178, accuracy: 1.0000)
Epoch [20/20]   34 time: 73.97s, d_loss: 1.5545 g_loss: 0.0284 (mse: 0.
0134, psnr: 27.3336, accuracy: 0.5000)
Epoch [20/20]   35 time: 73.70s, d_loss: 1.0919 g_loss: 0.0276 (mse: 0.
0101, psnr: 27.5685, accuracy: 0.5000)
Epoch [20/20]   36 time: 70.58s, d_loss: 1.7870 g_loss: 0.0234 (mse: 0.
0078, psnr: 28.0742, accuracy: 0.5000)
Epoch [20/20]   37 time: 67.91s, d_loss: 0.3188 g_loss: 0.0268 (mse: 0.
0088, psnr: 27.6284, accuracy: 1.0000)
Epoch [20/20]   38 time: 72.48s, d_loss: 0.6380 g_loss: 0.0254 (mse: 0.
0098, psnr: 27.5046, accuracy: 0.7344)
Epoch [20/20]   39 time: 74.72s, d_loss: 0.5095 g_loss: 0.0285 (mse: 0.
0106, psnr: 26.8781, accuracy: 0.9688)
Epoch [20/20]   40 time: 75.31s, d_loss: 0.4060 g_loss: 0.0290 (mse: 0.
0105, psnr: 26.5334, accuracy: 1.0000)
Epoch [20/20]   41 time: 73.77s, d_loss: 0.6002 g_loss: 0.0303 (mse: 0.
0106, psnr: 26.7029, accuracy: 0.7344)
Epoch [20/20]   42 time: 73.52s, d_loss: 0.6889 g_loss: 0.0302 (mse: 0.
0124, psnr: 25.9636, accuracy: 0.9844)
Epoch [20/20]   43 time: 75.32s, d_loss: 0.4679 g_loss: 0.0303 (mse: 0.
0109, psnr: 27.0150, accuracy: 1.0000)
Epoch [20/20]   44 time: 73.69s, d_loss: 0.7314 g_loss: 0.0282 (mse: 0.
0133, psnr: 26.9231, accuracy: 0.9844)
Epoch [20/20]   45 time: 71.98s, d_loss: 0.5500 g_loss: 0.0274 (mse: 0.
0118, psnr: 26.5463, accuracy: 1.0000)
Epoch [20/20]   46 time: 62.59s, d_loss: 1.0625 g_loss: 0.0226 (mse: 0.
0073, psnr: 28.2870, accuracy: 0.5179)
[*] Epoch: [20/20] time: 3350.55s, d_loss: 0.7275 g_loss: 0.0273 (mse:
 0.010266, psnr: 27.2099, accuracy: 0.8488)
[*] save images
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/g_srgan.npz
```

```
[TL] [*] Saved
[TL] [*] Saving TL params into ../output/SRGAN2/checkpoint/d_srgan.npz
[TL] [*] Saved
```

Prediction

In [3]:
```
test_lr_path = '../data/train_set/LR'
checkpoint_path = '../output/SRGAN2/checkpoint'
save_path = '../output/SRGAN2'
tf.reset_default_graph()
predict(test_lr_path=test_lr_path, checkpoint_path=checkpoint_path, save
_path=save_path)
```

```
[TL] [*] creates ../output/SRGAN2/test_gen ...
[TL] read 32 from ../data/train_set/LR
[TL] read 64 from ../data/train_set/LR
[TL] read 96 from ../data/train_set/LR
[TL] read 128 from ../data/train_set/LR
[TL] read 160 from ../data/train_set/LR
[TL] read 192 from ../data/train_set/LR
[TL] read 224 from ../data/train_set/LR
[TL] read 256 from ../data/train_set/LR
[TL] read 288 from ../data/train_set/LR
[TL] read 320 from ../data/train_set/LR
[TL] read 352 from ../data/train_set/LR
[TL] read 384 from ../data/train_set/LR
[TL] read 416 from ../data/train_set/LR
[TL] read 448 from ../data/train_set/LR
[TL] read 480 from ../data/train_set/LR
[TL] read 512 from ../data/train_set/LR
[TL] read 544 from ../data/train_set/LR
[TL] read 576 from ../data/train_set/LR
[TL] read 608 from ../data/train_set/LR
[TL] read 640 from ../data/train_set/LR
[TL] read 672 from ../data/train_set/LR
[TL] read 704 from ../data/train_set/LR
[TL] read 736 from ../data/train_set/LR
[TL] read 768 from ../data/train_set/LR
[TL] read 800 from ../data/train_set/LR
[TL] read 832 from ../data/train_set/LR
[TL] read 864 from ../data/train_set/LR
[TL] read 896 from ../data/train_set/LR
[TL] read 928 from ../data/train_set/LR
[TL] read 960 from ../data/train_set/LR
[TL] read 992 from ../data/train_set/LR
[TL] read 1024 from ../data/train_set/LR
[TL] read 1056 from ../data/train_set/LR
[TL] read 1088 from ../data/train_set/LR
[TL] read 1120 from ../data/train_set/LR
[TL] read 1152 from ../data/train_set/LR
[TL] read 1184 from ../data/train_set/LR
[TL] read 1216 from ../data/train_set/LR
[TL] read 1248 from ../data/train_set/LR
[TL] read 1280 from ../data/train_set/LR
[TL] read 1312 from ../data/train_set/LR
[TL] read 1344 from ../data/train_set/LR
[TL] read 1376 from ../data/train_set/LR
[TL] read 1408 from ../data/train_set/LR
[TL] read 1440 from ../data/train_set/LR
[TL] read 1472 from ../data/train_set/LR
[TL] read 1500 from ../data/train_set/LR
[TL] InputLayer  SRGAN_g/in: (1, ?, ?, 3)
[TL] Conv2d SRGAN_g/n64s1/c: n_filter: 64 filter_size: (3, 3) strides:
(1, 1) pad: SAME act: relu
[TL] Conv2d SRGAN_g/n64s1/c1/0: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/0: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/0: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
```

```
[TL] BatchNormLayer SRGAN_g/n64s1/b2/0: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/0: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/1: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/1: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/1: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/1: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/1: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/2: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/2: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/2: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/2: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/2: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/3: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/3: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/3: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/3: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/3: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/4: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/4: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/4: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/4: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/4: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/5: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/5: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/5: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/5: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/5: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/6: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/6: decay: 0.900000 epsilon: 0.0000
```

```
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/6: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/6: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/6: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/7: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/7: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/7: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/7: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/7: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/8: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/8: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/8: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/8: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/8: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/9: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/9: decay: 0.900000 epsilon: 0.0000
10 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/9: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/9: decay: 0.900000 epsilon: 0.0000
10 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/9: size: (1, ?, ?, 64) fn:
add
[TL] Conv2d SRGAN_g/n64s1/c1/10: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/10: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/10: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/10: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/10: size: (1, ?, ?, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/11: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/11: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/11: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/11: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/11: size: (1, ?, ?, 64) f
n: add
```

```
[TL] Conv2d SRGAN_g/n64s1/c1/12: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/12: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/12: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/12: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/12: size: (1, ?, ?, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/13: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/13: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/13: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/13: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/13: size: (1, ?, ?, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/14: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/14: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/14: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/14: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/14: size: (1, ?, ?, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c1/15: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b1/15: decay: 0.900000 epsilon: 0.000
010 act: relu is_train: False
[TL] Conv2d SRGAN_g/n64s1/c2/15: n_filter: 64 filter_size: (3, 3) strid
es: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b2/15: decay: 0.900000 epsilon: 0.000
010 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/b_residual_add/15: size: (1, ?, ?, 64) f
n: add
[TL] Conv2d SRGAN_g/n64s1/c/m: n_filter: 64 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] BatchNormLayer SRGAN_g/n64s1/b/m: decay: 0.900000 epsilon: 0.00001
0 act: No Activation is_train: False
[TL] ElementwiseLayer SRGAN_g/add3: size: (1, ?, ?, 64) fn: add
[TL] Conv2d SRGAN_g/n256s1/1: n_filter: 256 filter_size: (3, 3) stride
s: (1, 1) pad: SAME act: No Activation
[TL] SubpixelConv2d  SRGAN_g/pixelshufflerx2/1: scale: 2 n_out_channel:
64 act: relu
[TL] Conv2d SRGAN_g/out: n_filter: 3 filter_size: (1, 1) strides: (1,
1) pad: SAME act: tanh
[TL] [*] Load ../output/SRGAN2/checkpoint/g_srgan.npz SUCCESS!
saving 10 images, ok
saving 20 images, ok
saving 30 images, ok
saving 40 images, ok
saving 50 images, ok
```

```
saving 60 images, ok
saving 70 images, ok
saving 80 images, ok
saving 90 images, ok
saving 100 images, ok
saving 110 images, ok
saving 120 images, ok
saving 130 images, ok
saving 140 images, ok
saving 150 images, ok
saving 160 images, ok
saving 170 images, ok
saving 180 images, ok
saving 190 images, ok
saving 200 images, ok
saving 210 images, ok
saving 220 images, ok
saving 230 images, ok
saving 240 images, ok
saving 250 images, ok
saving 260 images, ok
saving 270 images, ok
saving 280 images, ok
saving 290 images, ok
saving 300 images, ok
saving 310 images, ok
saving 320 images, ok
saving 330 images, ok
saving 340 images, ok
saving 350 images, ok
saving 360 images, ok
saving 370 images, ok
saving 380 images, ok
saving 390 images, ok
saving 400 images, ok
saving 410 images, ok
saving 420 images, ok
saving 430 images, ok
saving 440 images, ok
saving 450 images, ok
saving 460 images, ok
saving 470 images, ok
saving 480 images, ok
saving 490 images, ok
saving 500 images, ok
saving 510 images, ok
saving 520 images, ok
saving 530 images, ok
saving 540 images, ok
saving 550 images, ok
saving 560 images, ok
saving 570 images, ok
saving 580 images, ok
saving 590 images, ok
saving 600 images, ok
saving 610 images, ok
saving 620 images, ok
```

```
saving 630 images, ok
saving 640 images, ok
saving 650 images, ok
saving 660 images, ok
saving 670 images, ok
saving 680 images, ok
saving 690 images, ok
saving 700 images, ok
saving 710 images, ok
saving 720 images, ok
saving 730 images, ok
saving 740 images, ok
saving 750 images, ok
saving 760 images, ok
saving 770 images, ok
saving 780 images, ok
saving 790 images, ok
saving 800 images, ok
saving 810 images, ok
saving 820 images, ok
saving 830 images, ok
saving 840 images, ok
saving 850 images, ok
saving 860 images, ok
saving 870 images, ok
saving 880 images, ok
saving 890 images, ok
saving 900 images, ok
saving 910 images, ok
saving 920 images, ok
saving 930 images, ok
saving 940 images, ok
saving 950 images, ok
saving 960 images, ok
saving 970 images, ok
saving 980 images, ok
saving 990 images, ok
saving 1000 images, ok
saving 1010 images, ok
saving 1020 images, ok
saving 1030 images, ok
saving 1040 images, ok
saving 1050 images, ok
saving 1060 images, ok
saving 1070 images, ok
saving 1080 images, ok
saving 1090 images, ok
saving 1100 images, ok
saving 1110 images, ok
saving 1120 images, ok
saving 1130 images, ok
saving 1140 images, ok
saving 1150 images, ok
saving 1160 images, ok
saving 1170 images, ok
saving 1180 images, ok
saving 1190 images, ok
```

```
saving 1200 images, ok
saving 1210 images, ok
saving 1220 images, ok
saving 1230 images, ok
saving 1240 images, ok
saving 1250 images, ok
saving 1260 images, ok
saving 1270 images, ok
saving 1280 images, ok
saving 1290 images, ok
saving 1300 images, ok
saving 1310 images, ok
saving 1320 images, ok
saving 1330 images, ok
saving 1340 images, ok
saving 1350 images, ok
saving 1360 images, ok
saving 1370 images, ok
saving 1380 images, ok
saving 1390 images, ok
saving 1400 images, ok
saving 1410 images, ok
saving 1420 images, ok
saving 1430 images, ok
saving 1440 images, ok
saving 1450 images, ok
saving 1460 images, ok
saving 1470 images, ok
saving 1480 images, ok
saving 1490 images, ok
take: 3258.17s
```

In [ ]:

Generate images by bicubic

In [4]:
```
import tensorlayer as tl
import scipy
```

In [7]:
```
test_lr_path = '../data/train_set/LR'
test_lr_img_list = sorted(tl.files.load_file_list(path=test_lr_path, reg
x='.*.jpg', printable=False))
save_path = '../output/SRGAN2/test_bicubic'

imgs = tl.vis.read_images(test_lr_img_list, path=test_lr_path)
out = [scipy.misc.imresize(img, [img.shape[0]*2, img.shape[1]*2], interp
='bicubic', mode=None) for img in imgs]
for i in range(len(out)):
    tl.vis.save_image(out[i], os.path.join(save_path, '{}'.format(test_l
r_img_list[i])))
```

```
[TL] read 10 from ../data/train_set/LR
[TL] read 20 from ../data/train_set/LR
[TL] read 30 from ../data/train_set/LR
[TL] read 40 from ../data/train_set/LR
[TL] read 50 from ../data/train_set/LR
[TL] read 60 from ../data/train_set/LR
[TL] read 70 from ../data/train_set/LR
[TL] read 80 from ../data/train_set/LR
[TL] read 90 from ../data/train_set/LR
[TL] read 100 from ../data/train_set/LR
[TL] read 110 from ../data/train_set/LR
[TL] read 120 from ../data/train_set/LR
[TL] read 130 from ../data/train_set/LR
[TL] read 140 from ../data/train_set/LR
[TL] read 150 from ../data/train_set/LR
[TL] read 160 from ../data/train_set/LR
[TL] read 170 from ../data/train_set/LR
[TL] read 180 from ../data/train_set/LR
[TL] read 190 from ../data/train_set/LR
[TL] read 200 from ../data/train_set/LR
[TL] read 210 from ../data/train_set/LR
[TL] read 220 from ../data/train_set/LR
[TL] read 230 from ../data/train_set/LR
[TL] read 240 from ../data/train_set/LR
[TL] read 250 from ../data/train_set/LR
[TL] read 260 from ../data/train_set/LR
[TL] read 270 from ../data/train_set/LR
[TL] read 280 from ../data/train_set/LR
[TL] read 290 from ../data/train_set/LR
[TL] read 300 from ../data/train_set/LR
[TL] read 310 from ../data/train_set/LR
[TL] read 320 from ../data/train_set/LR
[TL] read 330 from ../data/train_set/LR
[TL] read 340 from ../data/train_set/LR
[TL] read 350 from ../data/train_set/LR
[TL] read 360 from ../data/train_set/LR
[TL] read 370 from ../data/train_set/LR
[TL] read 380 from ../data/train_set/LR
[TL] read 390 from ../data/train_set/LR
[TL] read 400 from ../data/train_set/LR
[TL] read 410 from ../data/train_set/LR
[TL] read 420 from ../data/train_set/LR
[TL] read 430 from ../data/train_set/LR
[TL] read 440 from ../data/train_set/LR
[TL] read 450 from ../data/train_set/LR
[TL] read 460 from ../data/train_set/LR
[TL] read 470 from ../data/train_set/LR
[TL] read 480 from ../data/train_set/LR
[TL] read 490 from ../data/train_set/LR
[TL] read 500 from ../data/train_set/LR
[TL] read 510 from ../data/train_set/LR
[TL] read 520 from ../data/train_set/LR
[TL] read 530 from ../data/train_set/LR
[TL] read 540 from ../data/train_set/LR
[TL] read 550 from ../data/train_set/LR
[TL] read 560 from ../data/train_set/LR
[TL] read 570 from ../data/train_set/LR
```

```
[TL] read 580 from ../data/train_set/LR
[TL] read 590 from ../data/train_set/LR
[TL] read 600 from ../data/train_set/LR
[TL] read 610 from ../data/train_set/LR
[TL] read 620 from ../data/train_set/LR
[TL] read 630 from ../data/train_set/LR
[TL] read 640 from ../data/train_set/LR
[TL] read 650 from ../data/train_set/LR
[TL] read 660 from ../data/train_set/LR
[TL] read 670 from ../data/train_set/LR
[TL] read 680 from ../data/train_set/LR
[TL] read 690 from ../data/train_set/LR
[TL] read 700 from ../data/train_set/LR
[TL] read 710 from ../data/train_set/LR
[TL] read 720 from ../data/train_set/LR
[TL] read 730 from ../data/train_set/LR
[TL] read 740 from ../data/train_set/LR
[TL] read 750 from ../data/train_set/LR
[TL] read 760 from ../data/train_set/LR
[TL] read 770 from ../data/train_set/LR
[TL] read 780 from ../data/train_set/LR
[TL] read 790 from ../data/train_set/LR
[TL] read 800 from ../data/train_set/LR
[TL] read 810 from ../data/train_set/LR
[TL] read 820 from ../data/train_set/LR
[TL] read 830 from ../data/train_set/LR
[TL] read 840 from ../data/train_set/LR
[TL] read 850 from ../data/train_set/LR
[TL] read 860 from ../data/train_set/LR
[TL] read 870 from ../data/train_set/LR
[TL] read 880 from ../data/train_set/LR
[TL] read 890 from ../data/train_set/LR
[TL] read 900 from ../data/train_set/LR
[TL] read 910 from ../data/train_set/LR
[TL] read 920 from ../data/train_set/LR
[TL] read 930 from ../data/train_set/LR
[TL] read 940 from ../data/train_set/LR
[TL] read 950 from ../data/train_set/LR
[TL] read 960 from ../data/train_set/LR
[TL] read 970 from ../data/train_set/LR
[TL] read 980 from ../data/train_set/LR
[TL] read 990 from ../data/train_set/LR
[TL] read 1000 from ../data/train_set/LR
[TL] read 1010 from ../data/train_set/LR
[TL] read 1020 from ../data/train_set/LR
[TL] read 1030 from ../data/train_set/LR
[TL] read 1040 from ../data/train_set/LR
[TL] read 1050 from ../data/train_set/LR
[TL] read 1060 from ../data/train_set/LR
[TL] read 1070 from ../data/train_set/LR
[TL] read 1080 from ../data/train_set/LR
[TL] read 1090 from ../data/train_set/LR
[TL] read 1100 from ../data/train_set/LR
[TL] read 1110 from ../data/train_set/LR
[TL] read 1120 from ../data/train_set/LR
[TL] read 1130 from ../data/train_set/LR
[TL] read 1140 from ../data/train_set/LR
```

```
[TL] read 1150 from ../data/train_set/LR
[TL] read 1160 from ../data/train_set/LR
[TL] read 1170 from ../data/train_set/LR
[TL] read 1180 from ../data/train_set/LR
[TL] read 1190 from ../data/train_set/LR
[TL] read 1200 from ../data/train_set/LR
[TL] read 1210 from ../data/train_set/LR
[TL] read 1220 from ../data/train_set/LR
[TL] read 1230 from ../data/train_set/LR
[TL] read 1240 from ../data/train_set/LR
[TL] read 1250 from ../data/train_set/LR
[TL] read 1260 from ../data/train_set/LR
[TL] read 1270 from ../data/train_set/LR
[TL] read 1280 from ../data/train_set/LR
[TL] read 1290 from ../data/train_set/LR
[TL] read 1300 from ../data/train_set/LR
[TL] read 1310 from ../data/train_set/LR
[TL] read 1320 from ../data/train_set/LR
[TL] read 1330 from ../data/train_set/LR
[TL] read 1340 from ../data/train_set/LR
[TL] read 1350 from ../data/train_set/LR
[TL] read 1360 from ../data/train_set/LR
[TL] read 1370 from ../data/train_set/LR
[TL] read 1380 from ../data/train_set/LR
[TL] read 1390 from ../data/train_set/LR
[TL] read 1400 from ../data/train_set/LR
[TL] read 1410 from ../data/train_set/LR
[TL] read 1420 from ../data/train_set/LR
[TL] read 1430 from ../data/train_set/LR
[TL] read 1440 from ../data/train_set/LR
[TL] read 1450 from ../data/train_set/LR
[TL] read 1460 from ../data/train_set/LR
[TL] read 1470 from ../data/train_set/LR
[TL] read 1480 from ../data/train_set/LR
[TL] read 1490 from ../data/train_set/LR
[TL] read 1500 from ../data/train_set/LR

/Users/james/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.p
y:6: DeprecationWarning: `imresize` is deprecated!
`imresize` is deprecated in SciPy 1.0.0, and will be removed in 1.2.0.
Use ``skimage.transform.resize`` instead.
```

Comparison by psnr

In [1]:
```
import numpy as np
import tensorlayer as tl
```

In [2]:
```
test_hr_img_path = '../data/train_set/HR'
gen_hr_img_path = '../output/SRGAN2/test_srgan'
def psnr(img1, img2):
    '''
    img1 and img2 are two 3-dimention images
    '''
    return 10*np.log10(255*255/(np.square(img1-img2).mean()))
```

In [3]:
```
test_hr_list = sorted(tl.files.load_file_list(path=test_hr_img_path, reg
x='.*.jpg', printable=False))
test_gen_list = sorted(tl.files.load_file_list(path=gen_hr_img_path, reg
x='.*.jpg', printable=False))

test_hr_imgs = tl.vis.read_images(test_hr_list, path=test_hr_img_path)
test_gen_imgs = tl.vis.read_images(test_gen_list, path=gen_hr_img_path)

# mean psnr
np.mean([psnr(img1,img2) for img1, img2 in zip(test_hr_imgs, test_gen_im
gs)])
```

```
[TL] read 10 from ../data/train_set/HR
[TL] read 20 from ../data/train_set/HR
[TL] read 30 from ../data/train_set/HR
[TL] read 40 from ../data/train_set/HR
[TL] read 50 from ../data/train_set/HR
[TL] read 60 from ../data/train_set/HR
[TL] read 70 from ../data/train_set/HR
[TL] read 80 from ../data/train_set/HR
[TL] read 90 from ../data/train_set/HR
[TL] read 100 from ../data/train_set/HR
[TL] read 110 from ../data/train_set/HR
[TL] read 120 from ../data/train_set/HR
[TL] read 130 from ../data/train_set/HR
[TL] read 140 from ../data/train_set/HR
[TL] read 150 from ../data/train_set/HR
[TL] read 160 from ../data/train_set/HR
[TL] read 170 from ../data/train_set/HR
[TL] read 180 from ../data/train_set/HR
[TL] read 190 from ../data/train_set/HR
[TL] read 200 from ../data/train_set/HR
[TL] read 210 from ../data/train_set/HR
[TL] read 220 from ../data/train_set/HR
[TL] read 230 from ../data/train_set/HR
[TL] read 240 from ../data/train_set/HR
[TL] read 250 from ../data/train_set/HR
[TL] read 260 from ../data/train_set/HR
[TL] read 270 from ../data/train_set/HR
[TL] read 280 from ../data/train_set/HR
[TL] read 290 from ../data/train_set/HR
[TL] read 300 from ../data/train_set/HR
[TL] read 310 from ../data/train_set/HR
[TL] read 320 from ../data/train_set/HR
[TL] read 330 from ../data/train_set/HR
[TL] read 340 from ../data/train_set/HR
[TL] read 350 from ../data/train_set/HR
[TL] read 360 from ../data/train_set/HR
[TL] read 370 from ../data/train_set/HR
[TL] read 380 from ../data/train_set/HR
[TL] read 390 from ../data/train_set/HR
[TL] read 400 from ../data/train_set/HR
[TL] read 410 from ../data/train_set/HR
[TL] read 420 from ../data/train_set/HR
[TL] read 430 from ../data/train_set/HR
[TL] read 440 from ../data/train_set/HR
[TL] read 450 from ../data/train_set/HR
[TL] read 460 from ../data/train_set/HR
[TL] read 470 from ../data/train_set/HR
[TL] read 480 from ../data/train_set/HR
[TL] read 490 from ../data/train_set/HR
[TL] read 500 from ../data/train_set/HR
[TL] read 510 from ../data/train_set/HR
[TL] read 520 from ../data/train_set/HR
[TL] read 530 from ../data/train_set/HR
[TL] read 540 from ../data/train_set/HR
[TL] read 550 from ../data/train_set/HR
[TL] read 560 from ../data/train_set/HR
[TL] read 570 from ../data/train_set/HR
```

```
[TL] read 580 from ../data/train_set/HR
[TL] read 590 from ../data/train_set/HR
[TL] read 600 from ../data/train_set/HR
[TL] read 610 from ../data/train_set/HR
[TL] read 620 from ../data/train_set/HR
[TL] read 630 from ../data/train_set/HR
[TL] read 640 from ../data/train_set/HR
[TL] read 650 from ../data/train_set/HR
[TL] read 660 from ../data/train_set/HR
[TL] read 670 from ../data/train_set/HR
[TL] read 680 from ../data/train_set/HR
[TL] read 690 from ../data/train_set/HR
[TL] read 700 from ../data/train_set/HR
[TL] read 710 from ../data/train_set/HR
[TL] read 720 from ../data/train_set/HR
[TL] read 730 from ../data/train_set/HR
[TL] read 740 from ../data/train_set/HR
[TL] read 750 from ../data/train_set/HR
[TL] read 760 from ../data/train_set/HR
[TL] read 770 from ../data/train_set/HR
[TL] read 780 from ../data/train_set/HR
[TL] read 790 from ../data/train_set/HR
[TL] read 800 from ../data/train_set/HR
[TL] read 810 from ../data/train_set/HR
[TL] read 820 from ../data/train_set/HR
[TL] read 830 from ../data/train_set/HR
[TL] read 840 from ../data/train_set/HR
[TL] read 850 from ../data/train_set/HR
[TL] read 860 from ../data/train_set/HR
[TL] read 870 from ../data/train_set/HR
[TL] read 880 from ../data/train_set/HR
[TL] read 890 from ../data/train_set/HR
[TL] read 900 from ../data/train_set/HR
[TL] read 910 from ../data/train_set/HR
[TL] read 920 from ../data/train_set/HR
[TL] read 930 from ../data/train_set/HR
[TL] read 940 from ../data/train_set/HR
[TL] read 950 from ../data/train_set/HR
[TL] read 960 from ../data/train_set/HR
[TL] read 970 from ../data/train_set/HR
[TL] read 980 from ../data/train_set/HR
[TL] read 990 from ../data/train_set/HR
[TL] read 1000 from ../data/train_set/HR
[TL] read 1010 from ../data/train_set/HR
[TL] read 1020 from ../data/train_set/HR
[TL] read 1030 from ../data/train_set/HR
[TL] read 1040 from ../data/train_set/HR
[TL] read 1050 from ../data/train_set/HR
[TL] read 1060 from ../data/train_set/HR
[TL] read 1070 from ../data/train_set/HR
[TL] read 1080 from ../data/train_set/HR
[TL] read 1090 from ../data/train_set/HR
[TL] read 1100 from ../data/train_set/HR
[TL] read 1110 from ../data/train_set/HR
[TL] read 1120 from ../data/train_set/HR
[TL] read 1130 from ../data/train_set/HR
[TL] read 1140 from ../data/train_set/HR
```

```
[TL] read 1150 from ../data/train_set/HR
[TL] read 1160 from ../data/train_set/HR
[TL] read 1170 from ../data/train_set/HR
[TL] read 1180 from ../data/train_set/HR
[TL] read 1190 from ../data/train_set/HR
[TL] read 1200 from ../data/train_set/HR
[TL] read 1210 from ../data/train_set/HR
[TL] read 1220 from ../data/train_set/HR
[TL] read 1230 from ../data/train_set/HR
[TL] read 1240 from ../data/train_set/HR
[TL] read 1250 from ../data/train_set/HR
[TL] read 1260 from ../data/train_set/HR
[TL] read 1270 from ../data/train_set/HR
[TL] read 1280 from ../data/train_set/HR
[TL] read 1290 from ../data/train_set/HR
[TL] read 1300 from ../data/train_set/HR
[TL] read 1310 from ../data/train_set/HR
[TL] read 1320 from ../data/train_set/HR
[TL] read 1330 from ../data/train_set/HR
[TL] read 1340 from ../data/train_set/HR
[TL] read 1350 from ../data/train_set/HR
[TL] read 1360 from ../data/train_set/HR
[TL] read 1370 from ../data/train_set/HR
[TL] read 1380 from ../data/train_set/HR
[TL] read 1390 from ../data/train_set/HR
[TL] read 1400 from ../data/train_set/HR
[TL] read 1410 from ../data/train_set/HR
[TL] read 1420 from ../data/train_set/HR
[TL] read 1430 from ../data/train_set/HR
[TL] read 1440 from ../data/train_set/HR
[TL] read 1450 from ../data/train_set/HR
[TL] read 1460 from ../data/train_set/HR
[TL] read 1470 from ../data/train_set/HR
[TL] read 1480 from ../data/train_set/HR
[TL] read 1490 from ../data/train_set/HR
[TL] read 1500 from ../data/train_set/HR
[TL] read 10 from ../output/SRGAN2/test_srgan
[TL] read 20 from ../output/SRGAN2/test_srgan
[TL] read 30 from ../output/SRGAN2/test_srgan
[TL] read 40 from ../output/SRGAN2/test_srgan
[TL] read 50 from ../output/SRGAN2/test_srgan
[TL] read 60 from ../output/SRGAN2/test_srgan
[TL] read 70 from ../output/SRGAN2/test_srgan
[TL] read 80 from ../output/SRGAN2/test_srgan
[TL] read 90 from ../output/SRGAN2/test_srgan
[TL] read 100 from ../output/SRGAN2/test_srgan
[TL] read 110 from ../output/SRGAN2/test_srgan
[TL] read 120 from ../output/SRGAN2/test_srgan
[TL] read 130 from ../output/SRGAN2/test_srgan
[TL] read 140 from ../output/SRGAN2/test_srgan
[TL] read 150 from ../output/SRGAN2/test_srgan
[TL] read 160 from ../output/SRGAN2/test_srgan
[TL] read 170 from ../output/SRGAN2/test_srgan
[TL] read 180 from ../output/SRGAN2/test_srgan
[TL] read 190 from ../output/SRGAN2/test_srgan
[TL] read 200 from ../output/SRGAN2/test_srgan
[TL] read 210 from ../output/SRGAN2/test_srgan
```

```
[TL] read 220 from ../output/SRGAN2/test_srgan
[TL] read 230 from ../output/SRGAN2/test_srgan
[TL] read 240 from ../output/SRGAN2/test_srgan
[TL] read 250 from ../output/SRGAN2/test_srgan
[TL] read 260 from ../output/SRGAN2/test_srgan
[TL] read 270 from ../output/SRGAN2/test_srgan
[TL] read 280 from ../output/SRGAN2/test_srgan
[TL] read 290 from ../output/SRGAN2/test_srgan
[TL] read 300 from ../output/SRGAN2/test_srgan
[TL] read 310 from ../output/SRGAN2/test_srgan
[TL] read 320 from ../output/SRGAN2/test_srgan
[TL] read 330 from ../output/SRGAN2/test_srgan
[TL] read 340 from ../output/SRGAN2/test_srgan
[TL] read 350 from ../output/SRGAN2/test_srgan
[TL] read 360 from ../output/SRGAN2/test_srgan
[TL] read 370 from ../output/SRGAN2/test_srgan
[TL] read 380 from ../output/SRGAN2/test_srgan
[TL] read 390 from ../output/SRGAN2/test_srgan
[TL] read 400 from ../output/SRGAN2/test_srgan
[TL] read 410 from ../output/SRGAN2/test_srgan
[TL] read 420 from ../output/SRGAN2/test_srgan
[TL] read 430 from ../output/SRGAN2/test_srgan
[TL] read 440 from ../output/SRGAN2/test_srgan
[TL] read 450 from ../output/SRGAN2/test_srgan
[TL] read 460 from ../output/SRGAN2/test_srgan
[TL] read 470 from ../output/SRGAN2/test_srgan
[TL] read 480 from ../output/SRGAN2/test_srgan
[TL] read 490 from ../output/SRGAN2/test_srgan
[TL] read 500 from ../output/SRGAN2/test_srgan
[TL] read 510 from ../output/SRGAN2/test_srgan
[TL] read 520 from ../output/SRGAN2/test_srgan
[TL] read 530 from ../output/SRGAN2/test_srgan
[TL] read 540 from ../output/SRGAN2/test_srgan
[TL] read 550 from ../output/SRGAN2/test_srgan
[TL] read 560 from ../output/SRGAN2/test_srgan
[TL] read 570 from ../output/SRGAN2/test_srgan
[TL] read 580 from ../output/SRGAN2/test_srgan
[TL] read 590 from ../output/SRGAN2/test_srgan
[TL] read 600 from ../output/SRGAN2/test_srgan
[TL] read 610 from ../output/SRGAN2/test_srgan
[TL] read 620 from ../output/SRGAN2/test_srgan
[TL] read 630 from ../output/SRGAN2/test_srgan
[TL] read 640 from ../output/SRGAN2/test_srgan
[TL] read 650 from ../output/SRGAN2/test_srgan
[TL] read 660 from ../output/SRGAN2/test_srgan
[TL] read 670 from ../output/SRGAN2/test_srgan
[TL] read 680 from ../output/SRGAN2/test_srgan
[TL] read 690 from ../output/SRGAN2/test_srgan
[TL] read 700 from ../output/SRGAN2/test_srgan
[TL] read 710 from ../output/SRGAN2/test_srgan
[TL] read 720 from ../output/SRGAN2/test_srgan
[TL] read 730 from ../output/SRGAN2/test_srgan
[TL] read 740 from ../output/SRGAN2/test_srgan
[TL] read 750 from ../output/SRGAN2/test_srgan
[TL] read 760 from ../output/SRGAN2/test_srgan
[TL] read 770 from ../output/SRGAN2/test_srgan
[TL] read 780 from ../output/SRGAN2/test_srgan
```

```
[TL] read 790 from ../output/SRGAN2/test_srgan
[TL] read 800 from ../output/SRGAN2/test_srgan
[TL] read 810 from ../output/SRGAN2/test_srgan
[TL] read 820 from ../output/SRGAN2/test_srgan
[TL] read 830 from ../output/SRGAN2/test_srgan
[TL] read 840 from ../output/SRGAN2/test_srgan
[TL] read 850 from ../output/SRGAN2/test_srgan
[TL] read 860 from ../output/SRGAN2/test_srgan
[TL] read 870 from ../output/SRGAN2/test_srgan
[TL] read 880 from ../output/SRGAN2/test_srgan
[TL] read 890 from ../output/SRGAN2/test_srgan
[TL] read 900 from ../output/SRGAN2/test_srgan
[TL] read 910 from ../output/SRGAN2/test_srgan
[TL] read 920 from ../output/SRGAN2/test_srgan
[TL] read 930 from ../output/SRGAN2/test_srgan
[TL] read 940 from ../output/SRGAN2/test_srgan
[TL] read 950 from ../output/SRGAN2/test_srgan
[TL] read 960 from ../output/SRGAN2/test_srgan
[TL] read 970 from ../output/SRGAN2/test_srgan
[TL] read 980 from ../output/SRGAN2/test_srgan
[TL] read 990 from ../output/SRGAN2/test_srgan
[TL] read 1000 from ../output/SRGAN2/test_srgan
[TL] read 1010 from ../output/SRGAN2/test_srgan
[TL] read 1020 from ../output/SRGAN2/test_srgan
[TL] read 1030 from ../output/SRGAN2/test_srgan
[TL] read 1040 from ../output/SRGAN2/test_srgan
[TL] read 1050 from ../output/SRGAN2/test_srgan
[TL] read 1060 from ../output/SRGAN2/test_srgan
[TL] read 1070 from ../output/SRGAN2/test_srgan
[TL] read 1080 from ../output/SRGAN2/test_srgan
[TL] read 1090 from ../output/SRGAN2/test_srgan
[TL] read 1100 from ../output/SRGAN2/test_srgan
[TL] read 1110 from ../output/SRGAN2/test_srgan
[TL] read 1120 from ../output/SRGAN2/test_srgan
[TL] read 1130 from ../output/SRGAN2/test_srgan
[TL] read 1140 from ../output/SRGAN2/test_srgan
[TL] read 1150 from ../output/SRGAN2/test_srgan
[TL] read 1160 from ../output/SRGAN2/test_srgan
[TL] read 1170 from ../output/SRGAN2/test_srgan
[TL] read 1180 from ../output/SRGAN2/test_srgan
[TL] read 1190 from ../output/SRGAN2/test_srgan
[TL] read 1200 from ../output/SRGAN2/test_srgan
[TL] read 1210 from ../output/SRGAN2/test_srgan
[TL] read 1220 from ../output/SRGAN2/test_srgan
[TL] read 1230 from ../output/SRGAN2/test_srgan
[TL] read 1240 from ../output/SRGAN2/test_srgan
[TL] read 1250 from ../output/SRGAN2/test_srgan
[TL] read 1260 from ../output/SRGAN2/test_srgan
[TL] read 1270 from ../output/SRGAN2/test_srgan
[TL] read 1280 from ../output/SRGAN2/test_srgan
[TL] read 1290 from ../output/SRGAN2/test_srgan
[TL] read 1300 from ../output/SRGAN2/test_srgan
[TL] read 1310 from ../output/SRGAN2/test_srgan
[TL] read 1320 from ../output/SRGAN2/test_srgan
[TL] read 1330 from ../output/SRGAN2/test_srgan
[TL] read 1340 from ../output/SRGAN2/test_srgan
[TL] read 1350 from ../output/SRGAN2/test_srgan
```

```
[TL] read 1360 from ../output/SRGAN2/test_srgan
[TL] read 1370 from ../output/SRGAN2/test_srgan
[TL] read 1380 from ../output/SRGAN2/test_srgan
[TL] read 1390 from ../output/SRGAN2/test_srgan
[TL] read 1400 from ../output/SRGAN2/test_srgan
[TL] read 1410 from ../output/SRGAN2/test_srgan
[TL] read 1420 from ../output/SRGAN2/test_srgan
[TL] read 1430 from ../output/SRGAN2/test_srgan
[TL] read 1440 from ../output/SRGAN2/test_srgan
[TL] read 1450 from ../output/SRGAN2/test_srgan
[TL] read 1460 from ../output/SRGAN2/test_srgan
[TL] read 1470 from ../output/SRGAN2/test_srgan
[TL] read 1480 from ../output/SRGAN2/test_srgan
[TL] read 1490 from ../output/SRGAN2/test_srgan
[TL] read 1500 from ../output/SRGAN2/test_srgan
```

Out[3]: 31.064676199109826

In [ ]: