DarioNet

Improve Tiny Images Classification Through Super Resolution (and vice versa)

Dario Ruggeri, Simone Ercolino,
Riccardo Ceccaroni, Romeo Lanzino

DATA SOURCE

Train

15% of ImageNet
Training set
(~195K images
~500x400, ~21Gb)

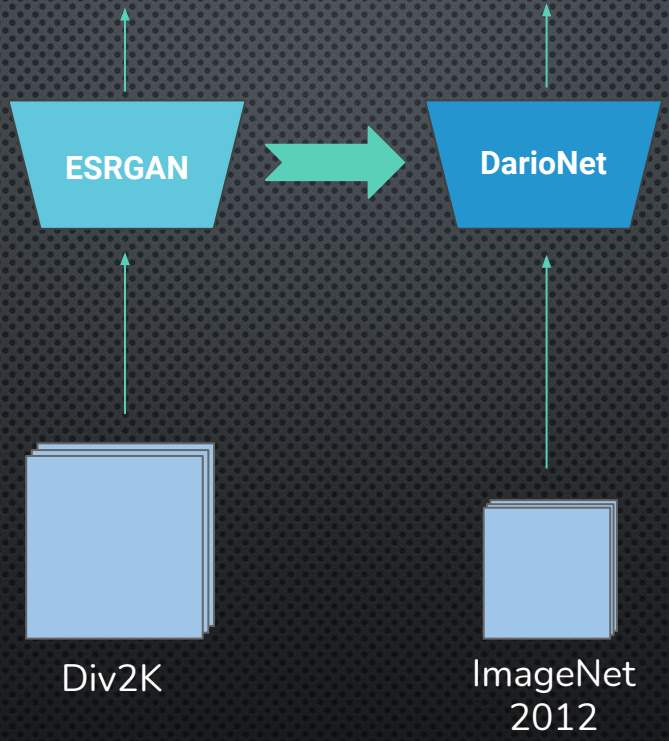
Validation

50% of ImageNet
Validation set
(~25K images ~500x400,
~3.5Gb)

Test

50% of ImageNet
Validation set
(~25K images ~500x400,
~3.5Gb)

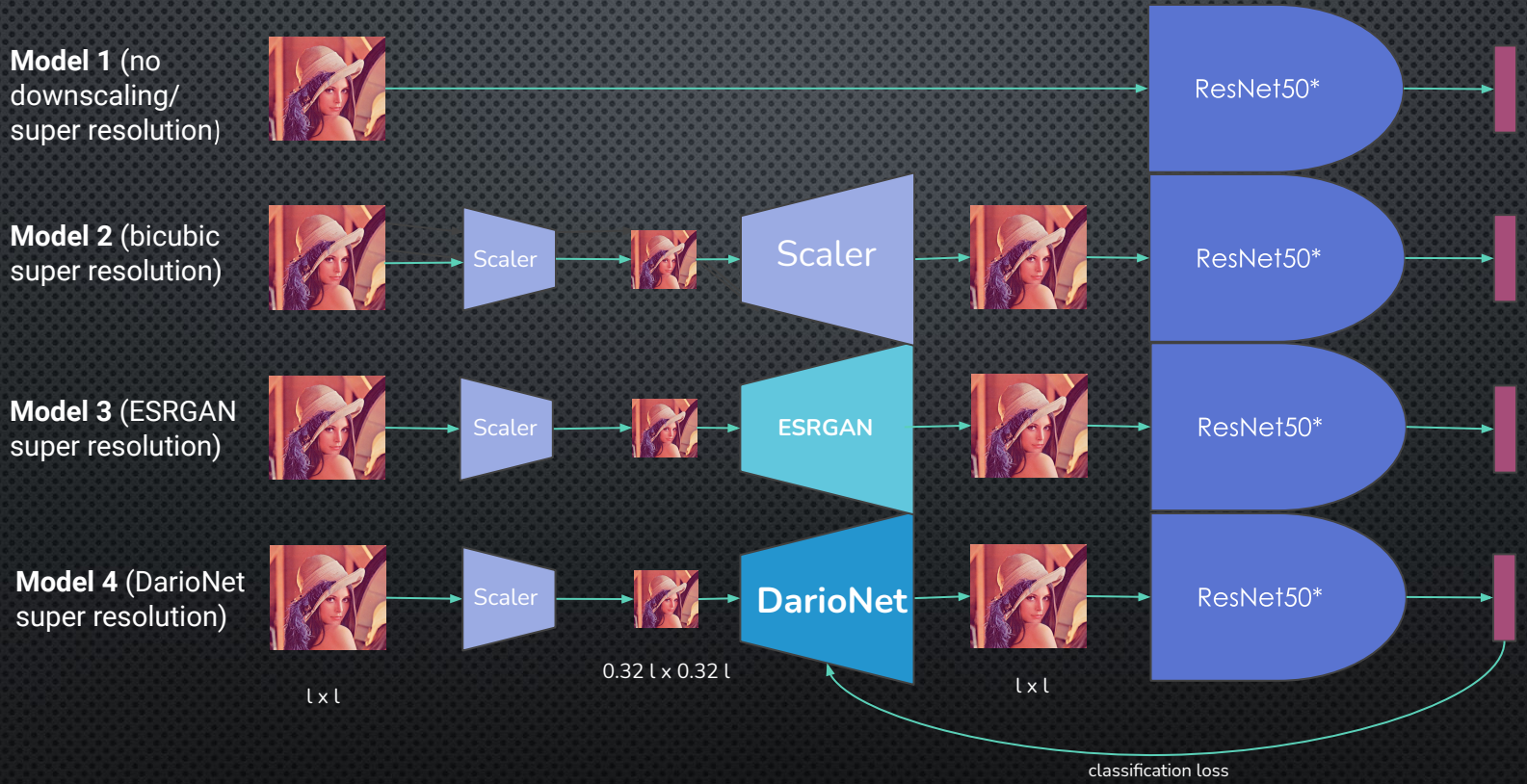
ESRGAN vs DarioNet



Images preprocessing

- **Data augmentation** seems to worsen the results and is not so needed since we have a very large dataset of images
- Previously we had random horizontal flip, random vertical flip and random crop at the end

End-to-end models pipelines

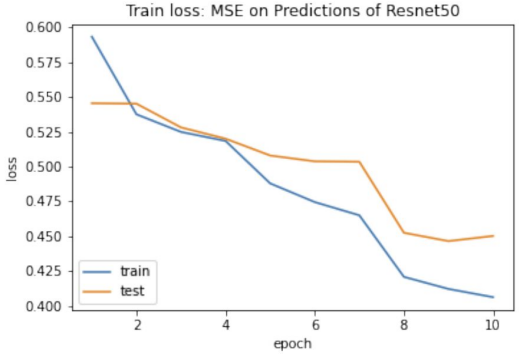


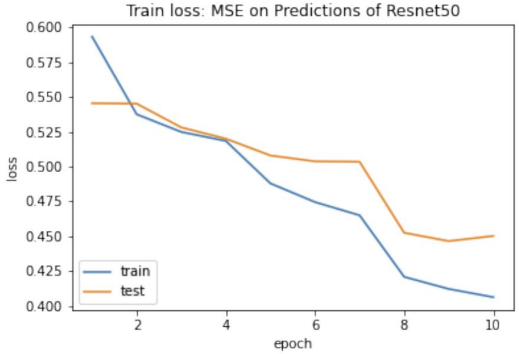
*previously ResNet18

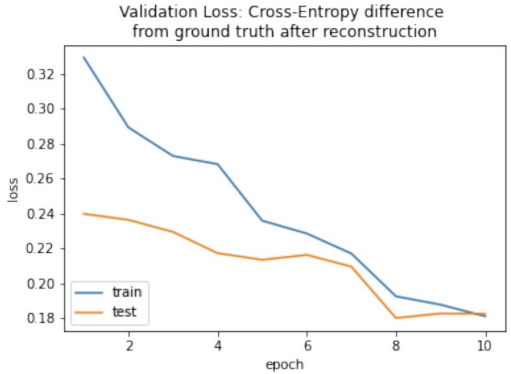
Losses (1/2)

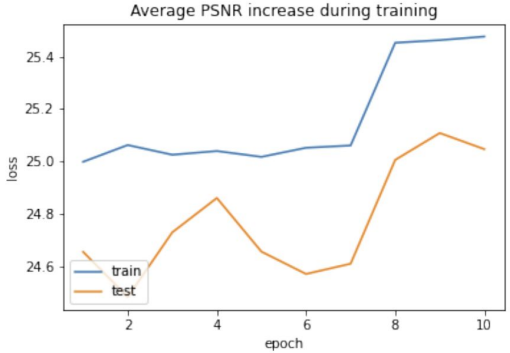
- **PSNR**
  - Indeed one of the most used loss for this application
  - considered unstable for training and seems to lead to blurred images
- **L1**
  - Manhattan distance
  - lead to blurred images
- **Classifier loss**
  - in our case, ResNet's loss
  - images not so beautiful for humans but emphasizing important features
  - highest scores
- **Combination of various losses**
  - tradeoff between aesthetics and performances
  - problem: different losses have different order of magnitude

Losses (2/2)


Train loss: MSE on Predictions of Resnet50
loss
epoch
train
test


Train loss: MSE on Predictions of Resnet50
loss
epoch
train
test


Validation Loss: Cross-Entropy difference
from ground truth after reconstruction
loss
epoch
train
test


Average PSNR increase during training
loss
epoch
train
test

Results (1/4)

Original image, predicted as 'church' with 58.87%
4x downscaled then 4x upscaled by DarioNet, predicted as 'church' with 79.85%
4x downscaled then 4x upscaled by ESRGAN, predicted as 'slot' with 60.66%
4x downscaled then upscaled with bilinear interpolation, predicted as 'monastery' with 21.4%
4x upscaled by DarioNet from the original, predicted as 'church' with 49.92%
4x upscaled by ESRGAN from the original, predicted as 'church' with 57.33%

Results (2/4)

Is it possible to play just right now with DarioNet:

- Apply Super Resolution to an image via Google Colab
- Take a look at test.py in our GitHub repo

Model 1 (no downscaling/ super resolution)

Model 2 (bilinear super resolution)

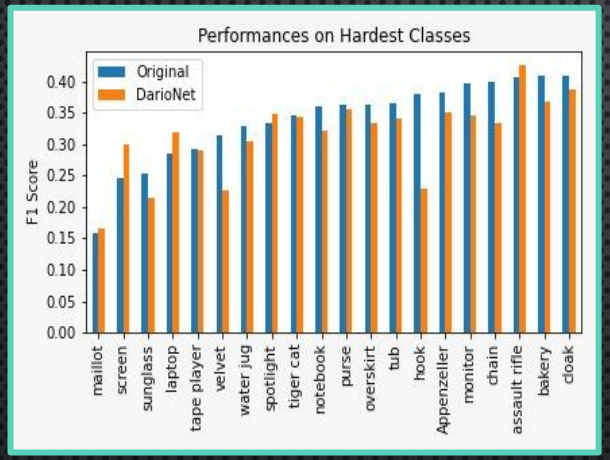
Model 3 (ESRGAN super resolution)

Model 4 (DarioNet super resolution)

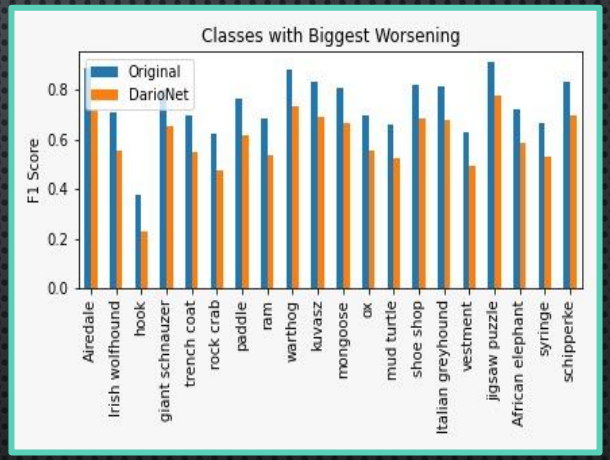
Results (3/4)

Name of the model	Average cross entropy loss	Average PSNR	Accuracy	Total time (test)
Model 1 (Original images)	0.9112	inf	0.7676	1.5 minutes
Model 2 (bilinear upscaling)	1.9864	25.2466 Db	0.5446	1.5 minutes
Model 3 (ESRGAN super resolution)	1.6946	22.5088 Db	0.6063	28 minutes
Model 4 (DarioNet super resolution)	1.0931	25.2211 Db	0.7275	28 minutes

Results (4/4)


Performances on Hardest Classes
Original
DarioNet
F1 Score
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
maillot
screen
sunglass
laptop
tape player
velvet
water jug
spotlight
tiger cat
notebook
purse
overskirt
tub
hook
Appenzeller
monitor
chain
assault rifle
bakery
cloak

Classes with Biggest Improvements
Original
DarioNet
F1 Score
0.0
0.2
0.4
0.6
0.8
clog
scabbard
iPod
potpie
mortarboard
lampshade
table lamp
beer glass
screen
bulletproof vest
electric locomotive
ashcan
dragonfly
breastplate
swab
digital watch
bib
reflex camera
fire screen
backpack


Classes with Biggest Worsening
Original
DarioNet
F1 Score
0.0
0.2
0.4
0.6
0.8
Airedale
Irish wolfhound
hook
giant schnauzer
trench coat
rock crab
paddle
ram
warthog
kuvasz
mongoose
ox
mud turtle
shoe shop
Italian greyhound
vestment
jigsaw puzzle
African elephant
syringe
schipperke

Conclusions and Future work

- **Starting idea**: evaluate the performances of super-resolution on different task.

- **Primary interest of our work**: improve the results on the classification task starting from low resolution inputs

- A step towards **image compression**

- **Train also ResNet50** (testing on tinyImagenet dataset)

- train DarioNet on **new tasks** like semantic segmentation or object detection

- See the results when classification has **less classes to predict**