

Alexnet_Verification_with_Imgnet_training

April 16, 2024

```
[1]: import numpy as np
from functools import partial
from typing import Any, Optional

import os
import cv2
import time

import pandas as pd
import torch.nn.init as init

import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import Dataset, DataLoader
from PIL import Image
from torchvision import transforms
# from torch.transforms._presets import ImageClassification
# from torch.utils import _log_api_usage_once
# from ._api import register_model, Weights, WeightsEnum
# from ._meta import _IMAGENET_CATEGORIES
# from ._utils import _ovewrite_named_param, handle_legacy_interface

model_alex_given = torch.hub.load('pytorch/vision:v0.10.0', 'alexnet',
    pretrained=True)
model_alex_given.eval()
# Device configuration
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
```

Using cache found in C:\Users\Limit\.cache\torch\hub\pytorch_vision_v0.10.0
C:\Apps installed by Lim\anaconda3\Lib\site-packages\torchvision\models_utils.py:208: UserWarning: The parameter 'pretrained' is deprecated since 0.13 and may be removed in the future, please use 'weights' instead.

```
warnings.warn(
C:\Apps installed by Lim\anaconda3\Lib\site-
```

packages\torchvision\models_utils.py:223: UserWarning: Arguments other than a weight enum or `None` for 'weights' are deprecated since 0.13 and may be removed in the future. The current behavior is equivalent to passing `weights=AlexNet_Weights.IMAGENET1K_V1`. You can also use `weights=AlexNet_Weights.DEFAULT` to get the most up-to-date weights.

```
warnings.warn(msg)
```

```
[2]: class AlexNet(nn.Module):
    def __init__(self, num_classes: int = 1000, dropout: float = 0.5) -> None:
        super().__init__()
        # _log_api_usage_once(self)
        self.features = nn.Sequential(
            nn.Conv2d(3, 96, kernel_size=11, stride=4, padding=2),
            nn.BatchNorm2d(96),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=3, stride=2),
            nn.Conv2d(96, 256, kernel_size=5, padding=2),
            nn.BatchNorm2d(256),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=3, stride=2),
            nn.Conv2d(256, 384, kernel_size=3, padding=1),
            nn.ReLU(inplace=True),
            nn.Conv2d(384, 384, kernel_size=3, padding=1),
            nn.ReLU(inplace=True),
            nn.Conv2d(384, 256, kernel_size=3, padding=1),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=3, stride=2),
        )
        self.avgpool = nn.AdaptiveAvgPool2d((6, 6))
        self.classifier = nn.Sequential(
            nn.Dropout(p=dropout),
            nn.Linear(256 * 6 * 6, 4096),
            nn.ReLU(inplace=True),
            nn.Dropout(p=dropout),
            nn.Linear(4096, 4096),
            nn.ReLU(inplace=True),
            nn.Linear(4096, num_classes),
        )

    def forward(self, x: torch.Tensor) -> torch.Tensor:
        x = self.features(x)
        x = self.avgpool(x)
        x = torch.flatten(x, 1)
        x = self.classifier(x)
        return x

    def initialize_weights(self):
```

```

for m in self.modules():
    if isinstance(m, nn.Conv2d) or isinstance(m, nn.Linear):
        # Initialize weights for convolutional and linear layers
        init.xavier_uniform_(m.weight)
        if m.bias is not None:
            # Initialize biases if they exist
            init.constant_(m.bias, 0)

```

```

[3]: class ConvertToRGB(object):
    def __call__(self, img):
        if img.shape[0] == 1: # Check if the image has only one channel
            img = torch.stack([img[0]] * 3, dim=0) # Convert single channel to RGB
        return img

```

```

preprocess = transforms.Compose([
    transforms.Resize(256),
    transforms.CenterCrop(224),
    transforms.ToTensor(),
    ConvertToRGB(),
    transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]),
])

```

```

[4]: image_path = "/Users/Limit/imagenet-object-localization-challenge_100"

```

```

filenames_image_path = []
label_train = []
root_image = []
counter = 0
current_label = 0
for root, _, filenames in os.walk(image_path):
    current_root = root
    for i in filenames:
        # print(i)
        counter += 1
        if ((counter) % 1300 == 0):
            current_label += 1
            print(counter)
            label_train.append(current_label)

    # print(current_root)
    # print(i)
    temp = current_root + "\\\" + i
    # print(temp)
    filenames_image_path.append(temp)

```

```

true_label = 0
correct_labels = 0
start_time = time.time()

counter_1=0
x_train = []
for i in range(26000):

    #     print(i)
    #     print(counter)
    image_name = filenames_image_path[i]
    # black and white
    #     image_name = "/Users/Limit/imagenet-object-localization-challenge/
    ↪n01440764/n01440764_15560.JPEG"
    input_image = Image.open(image_name)

    input_tensor = preprocess(input_image)
    #     input_batch = input_tensor.unsqueeze(0) # create a mini-batch as expected ↪
    ↪by the model
    input_batch = input_tensor
    # move the input and model to GPU for speed if available
    if torch.cuda.is_available():
        input_batch = input_batch.to('cuda')
        model_alex_given.to('cuda')
    x_train.append(input_batch)
    counter_1 += 1
    if ((counter_1+1) %1300 == 0):
        counter_1 += 1
        print(counter_1)

```

1300
 2600
 3900
 5200
 6500
 7800
 9100
 10400
 11700
 13000
 14300
 15600
 16900
 18200

19500
20800
22100
23400
24700
26000
27300
28600
29900
31200
32500
33800
35100
36400
37700
39000
40300
41600
42900
44200
45500
46800
48100
49400
50700
52000
53300
54600
55900
57200
58500
59800
61100
62400
63700
65000
66300
67600
68900
70200
71500
72800
74100
75400
76700
78000
79300
80600

81900
83200
84500
85800
87100
88400
89700
91000
92300
93600
94900
96200
97500
98800
100100
101400
102700
104000
105300
106600
107900
109200
110500
111800
113100
114400
115700
117000
118300
119600
120900
122200
123500
124800
126100
127400
128700
1300
2600
3900
5200
6500
7800
9100
10400
11700
13000
14300

15600
16900
18200
19500
20800
22100
23400
24700
26000

[]:

[5]: `y_train = label_train[:26000]`

[6]: `len(x_train)`

[6]: 26000

```
[7]: # # Add channel to greyscale images so that it has 3 channels required by  
    ↪ Alexnet.  
# # Add 1 more dimension to tensor to represent channel;  
# labels = []  
  
# # Function to load images from a folder directory with multiple sub-folders  
# def load_images_from_folder(folder):  
#     images = []  
#     for root, _, filenames in os.walk(folder):  
#         for filename in filenames:  
#             # here filename has the label information  
#             label_temp = filename.rsplit('_', 1)[0]  
#             labels.append(label_temp)  
#             img = cv2.imread(os.path.join(root, filename))  
#             if img is not None:  
#                 images.append(img)  
#     return images  
  
# image_path = "/Users/Limit/imagenet-object-localization-challenge/n01440764/"  
# images = load_images_from_folder(image_path)  
  
# # Ensure grayscale images have 3 channels  
# for idx, image in enumerate(images):  
#     if len(image.shape) == 2: # If grayscale image  
#         image = cv2.cvtColor(image, cv2.COLOR_GRAY2RGB) # Convert to  
    ↪ 3-channel image  
#         images[idx] = image
```

```

# # print the number of unique labels:
# # unique_elements = list(set(labels))
# # print(len(unique_elements))

```

```
[8]: len(x_train)
```

```
[8]: 26000
```

```

[9]: # Define your dataset class
class CustomDataset(Dataset):
    def __init__(self, data, labels):
        self.data = data
        self.labels = labels

    def __len__(self):
        return len(self.data)

    def __getitem__(self, idx):
        return self.data[idx], torch.tensor(self.labels[idx])

```

```

[10]: # unique_labels = labels.copy()
# unique_labels = list(set(unique_labels))
# unique_labels.sort()
# dict_labels = {}
# for i in range(len(unique_labels)):
#     dict_labels[unique_labels[i]] = i

```

```

[11]: # labels_numerical = []
# for i in labels:
#     labels_numerical.append(dict_labels[i])

```

```
[ ]:
```

```

[12]: # Define your loss function
criterion = nn.CrossEntropyLoss()

# Define your optimizer
model = AlexNet().to(device)
model.initialize_weights()

optimizer = optim.SGD(model.parameters(), lr=0.005, weight_decay=5e-4, momentum↵
    ↵ 0.9)

# Prepare your data
train_data = x_train # List of input tensors
train_labels = y_train # List of corresponding labels

```



```

dataset = CustomDataset(train_data, train_labels)
dataloader = DataLoader(dataset, batch_size=128, shuffle=True)

# Train your models
num_epochs = 45
for epoch in range(num_epochs):
    current_loss = 0.0
    for inputs, labels in dataloader:

        optimizer.zero_grad()
        outputs = model(inputs)
        Before = list(model.parameters())[0].clone()

        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()

        current_loss += loss.item()
        After = list(model.parameters())[0].clone()

    # Verify whether gradient is computed successfully
    print(torch.equal(Before.data, After.data))
    print(f'Epoch {epoch+1} finished')
    epoch_loss = current_loss / len(dataset)
    print(f'Epoch [{epoch+1}/{num_epochs}], Loss: {loss.item():.4f}')
    print()

```

False
Epoch 1 finished
Epoch [1/45], Loss: 2.0032

False
Epoch 2 finished
Epoch [2/45], Loss: 1.9860

False
Epoch 3 finished
Epoch [3/45], Loss: 1.2927

False
Epoch 4 finished
Epoch [4/45], Loss: 1.0513

False
Epoch 5 finished
Epoch [5/45], Loss: 0.9547

False

Epoch 6 finished
Epoch [6/45], Loss: 1.0239

False
Epoch 7 finished
Epoch [7/45], Loss: 1.9215

False
Epoch 8 finished
Epoch [8/45], Loss: 2.2980

False
Epoch 9 finished
Epoch [9/45], Loss: 1.1072

False
Epoch 10 finished
Epoch [10/45], Loss: 0.7342

False
Epoch 11 finished
Epoch [11/45], Loss: 0.8006

False
Epoch 12 finished
Epoch [12/45], Loss: 0.7885

False
Epoch 13 finished
Epoch [13/45], Loss: 1.2186

False
Epoch 14 finished
Epoch [14/45], Loss: 0.8843

False
Epoch 15 finished
Epoch [15/45], Loss: 0.4122

False
Epoch 16 finished
Epoch [16/45], Loss: 0.3417

False
Epoch 17 finished
Epoch [17/45], Loss: 0.4188

False

Epoch 18 finished
Epoch [18/45], Loss: 0.6130

False
Epoch 19 finished
Epoch [19/45], Loss: 0.6991

False
Epoch 20 finished
Epoch [20/45], Loss: 0.5701

False
Epoch 21 finished
Epoch [21/45], Loss: 0.6119

False
Epoch 22 finished
Epoch [22/45], Loss: 0.7217

False
Epoch 23 finished
Epoch [23/45], Loss: 0.3316

False
Epoch 24 finished
Epoch [24/45], Loss: 0.3394

False
Epoch 25 finished
Epoch [25/45], Loss: 0.4441

False
Epoch 26 finished
Epoch [26/45], Loss: 0.1296

False
Epoch 27 finished
Epoch [27/45], Loss: 0.1412

False
Epoch 28 finished
Epoch [28/45], Loss: 0.1402

False
Epoch 29 finished
Epoch [29/45], Loss: 0.1644

False

Epoch 30 finished
Epoch [30/45], Loss: 0.4472

False
Epoch 31 finished
Epoch [31/45], Loss: 0.3881

False
Epoch 32 finished
Epoch [32/45], Loss: 0.4345

False
Epoch 33 finished
Epoch [33/45], Loss: 0.4057

False
Epoch 34 finished
Epoch [34/45], Loss: 0.4787

False
Epoch 35 finished
Epoch [35/45], Loss: 0.0209

False
Epoch 36 finished
Epoch [36/45], Loss: 0.2424

False
Epoch 37 finished
Epoch [37/45], Loss: 0.1945

False
Epoch 38 finished
Epoch [38/45], Loss: 0.1105

False
Epoch 39 finished
Epoch [39/45], Loss: 0.2326

False
Epoch 40 finished
Epoch [40/45], Loss: 0.0905

False
Epoch 41 finished
Epoch [41/45], Loss: 0.5416

False

```
Epoch 42 finished  
Epoch [42/45], Loss: 0.2154
```

```
False  
Epoch 43 finished  
Epoch [43/45], Loss: 0.6643
```

```
False  
Epoch 44 finished  
Epoch [44/45], Loss: 0.3025
```

```
False  
Epoch 45 finished  
Epoch [45/45], Loss: 0.1706
```

```
[ ]:
```

```
[18]: labels_path = '/Users/Limit/imagenet_annot/validation_set_labels.csv'  
  
labels_df = pd.read_csv(labels_path)  
  
labels_df_leq_20 = labels_df[labels_df['label'] <= 20]  
  
labels_validation_images = labels_df_leq_20['label'].tolist()  
  
[23]: len(labels_df_leq_20['ImageId'].tolist())  
model.eval()
```

```
[23]: AlexNet(  
    (features): Sequential(  
      (0): Conv2d(3, 96, kernel_size=(11, 11), stride=(4, 4), padding=(2, 2))  
      (1): BatchNorm2d(96, eps=1e-05, momentum=0.1, affine=True,  
track_running_stats=True)  
      (2): ReLU(inplace=True)  
      (3): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,  
ceil_mode=False)  
      (4): Conv2d(96, 256, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))  
      (5): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,  
track_running_stats=True)  
      (6): ReLU(inplace=True)  
      (7): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,  
ceil_mode=False)  
      (8): Conv2d(256, 384, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))  
      (9): ReLU(inplace=True)  
      (10): Conv2d(384, 384, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))  
      (11): ReLU(inplace=True)  
      (12): Conv2d(384, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
```

```

        (13): ReLU(inplace=True)
        (14): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
    )
    (avgpool): AdaptiveAvgPool2d(output_size=(6, 6))
    (classifier): Sequential(
      (0): Dropout(p=0.5, inplace=False)
      (1): Linear(in_features=9216, out_features=4096, bias=True)
      (2): ReLU(inplace=True)
      (3): Dropout(p=0.5, inplace=False)
      (4): Linear(in_features=4096, out_features=4096, bias=True)
      (5): ReLU(inplace=True)
      (6): Linear(in_features=4096, out_features=1000, bias=True)
    )
  )
)

```

```

[24]: image_path = "/Users/Limit/imagenet-object-localization-challenge_validation/
↪val"
filenames_image_path = []
for root, _, filenames in os.walk(image_path):
    for i in filenames:
        #         print(i)
        if (i.split('.')[0] in labels_df_leq_20['ImageId'].tolist()):
            filenames_image_path.append(i)
true_label = 0
counter = 0
correct_labels = 0
start_time = time.time()
for i in range(len(filenames_image_path)):
    counter +=1
    #         print(i)
    #         print(counter)
    image_name = image_path + '/' + filenames_image_path[i]
    # black and white
    #         image_name = "/Users/Limit/imagenet-object-localization-challenge/
↪n01440764/n01440764_15560.JPEG"
    input_image = Image.open(image_name)

    input_tensor = preprocess(input_image)
    input_batch = input_tensor.unsqueeze(0) # create a mini-batch as expected ↵
↪by the model

    # move the input and model to GPU for speed if available
    if torch.cuda.is_available():
        input_batch = input_batch.to('cuda')

```

```

        model.to('cuda')

    if (counter%100 == 0):
        print("currently at", counter, 'current time is', time.time() -
↪start_time)
        with torch.no_grad():
            output = model(input_batch)
            # Tensor of shape 1000, with confidence scores over ImageNet's 1000 classes
            # if (torch.argmax(output[0]).item() == true_label):
            #     correct_labels += 1
            if (torch.argmax(output[0]).item() == labels_validation_images[i]):
                print('yes')
                correct_labels += 1
            else:
                print('no')
            # The output has unnormalized scores. To get probabilities, you can run a
↪softmax on it.
            probabilities = torch.nn.functional.softmax(output[0], dim=0)
            # print(probabilities)

print('training error on class 0 is')
print(correct_labels/counter)

```

```

yes
yes
no
yes
yes
no
yes
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
no
no
no
yes
no
yes
yes

```

no
yes
no
yes
yes
yes
yes
yes
no
yes
yes
no
no
yes
yes
yes
no
yes
yes
no
yes
yes
no
no
no
no
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
no
no
no
yes
yes
yes
no
yes
yes
yes
yes

no
no
yes
yes
yes
yes
yes
no
yes
no
yes
no
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
no
yes
yes
yes
currently at 100 current time is 2.5100786685943604
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
no
no
yes
yes
no
yes
no
yes

[illegible]

yes
no
no
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
yes
yes
no
no
yes
yes
yes
yes
no
no
yes
no
yes
yes
no
yes
yes
no
no
currently at 200 current time is 5.134443283081055
no
yes
yes
no
yes
yes
yes
no
yes
yes
yes
no
yes
no
yes

no
yes
no
yes
yes
yes
no
yes
yes
no
yes
yes
yes
no
yes
no
no
no
yes
yes
yes
yes
yes
yes
yes
no
yes
no
yes
yes
yes
no
no
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes

yes
yes
no
yes
yes
yes
yes
no
no
yes
no
yes
yes
yes
no
no
yes
no
no
yes
no
yes
yes
no
no
yes
yes
no
yes
no
no
no
yes
no
no
yes
yes
currently at 300 current time is 7.602491855621338
no
yes
yes
yes
yes
yes
yes
yes
yes
yes

yes
no
yes
no
no
no
yes
yes
yes
yes
no
yes
no
yes
yes
no
yes
no
no
yes
yes
yes
yes
yes
yes
no
yes
yes
no
yes
yes
no
no
yes
no
yes
yes
yes
yes
yes
no
yes
yes
yes
no
no
no

yes
no
yes
yes
no
yes
yes
yes
no
no
yes
yes
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
no
yes
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
currently at 400 current time is 10.127880811691284
no
yes
no
no
yes

no
yes
no
no
yes
yes
yes
yes
no
no
no
no
no
yes
yes
no
yes
yes
yes
yes
yes
no
yes
no
no
yes
no
yes
yes
yes
no
yes
no
yes
yes
no
yes
no
yes
no
no
no
no
yes
yes
no

yes
no
yes
no
yes
no
yes
yes
no
no
no
yes
yes
no
yes
no
yes
no
no
yes
no
no
yes
no
yes
yes
no
yes
no
no
no
yes
yes
yes
yes
yes
no
yes
no
no
yes
currently at 500 current time is 12.701905250549316

yes
yes
yes
yes
no
yes
yes
no
yes
yes
yes
yes
yes
yes
no
no
yes
no
no
yes
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
no
yes
yes
no
no
yes
yes

yes
yes
yes
yes
yes
yes
no
yes
no
no
yes
yes
no
yes
no
yes
no
no
yes
yes
yes
yes
yes
yes
yes
yes
no
no
yes
yes
yes
yes
yes
no
no
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
no
yes
yes
yes
no

[illegible]

yes
no
yes
no
yes
no
yes
no
yes
yes
no
yes
yes
yes
no
yes
yes
yes
yes
yes
no
yes
yes
no
yes
yes
no
yes
yes
no
yes
yes

yes
no
no
no
yes
yes
yes
yes
yes
currently at 700 current time is 17.628112316131592
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
yes
no
no
yes
yes
yes
no
yes
no
yes
yes
yes
no
yes
no
yes
no
yes

yes
no
yes
yes
no
yes
yes
yes
no
yes
yes
yes
yes
no
yes
yes
no
no
no
yes
no
no
yes
yes
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
yes
no
no
no
no
yes
no
yes
yes
no
yes
yes

yes
yes
yes
no
yes
yes
yes
yes
no
yes
no
yes
currently at 800 current time is 20.075451135635376
yes
yes
yes
yes
yes
no
no
yes
no
no
yes
no
no
yes
no
yes
yes
yes
yes
no
yes
no
yes
yes
yes
yes
no
yes
yes
no
yes
yes
yes
yes
yes

yes
yes
yes
yes
no
yes
yes
no
yes
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
yes
yes
no
yes
no
no
yes
no
yes
yes
yes
yes
yes
yes
no
yes
no
no
yes
no
no
yes
yes

no
yes
yes
no
yes
yes
no
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
no
yes
currently at 900 current time is 22.549293994903564
yes
yes
no
no
no
no
no
no
yes
no
no
yes
yes
yes
yes
yes
no
no
no
no
no
yes
yes
no
yes
yes
no
no

yes
yes
yes
yes
yes
no
yes
yes
yes
no
yes
yes
yes
yes
yes
no
yes
no
yes
no
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
no
yes
no
yes
no
yes
yes
yes
yes
no
yes
no
yes
no
yes
yes
yes
yes

no
yes
yes
no
no
yes
yes
no
yes
no
no
yes
yes
no
yes
yes
no
no
yes
no
yes
yes
yes
yes
currently at 1000 current time is 25.02297043800354
yes
yes
yes
yes
yes
no
no
yes
no
yes
no
no
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
yes
yes

no
no
no
no
yes
yes
yes
yes
yes
yes
yes
yes
no
yes
yes
yes
yes
no
yes
yes
yes
no
no
no
yes
yes
yes
training error on class 0 is
0.6752380952380952

```
[16]: # OLD code

# model_alex_given.eval()

# for i in range(40):
#     input_batch = input_tensor[i].unsqueeze(0) # create a mini-batch as
#     ↪ expected by the model

#     prediction = model_alex_given(input_batch)
#     print('prediction:')
#     print(torch.argmax(prediction))
#     print("answer:")
#     print(labels_numerical[i])
#     print()
```

```
[22]: torch.save(model.state_dict(), 'model_weights_alexnet_45_epoch.pth')
```

```
[17]: filenames_image_path
```

```
[17]: ['ILSVRC2012_val_00000198.JPEG',  
      'ILSVRC2012_val_00000217.JPEG',  
      'ILSVRC2012_val_00000236.JPEG',  
      'ILSVRC2012_val_00000247.JPEG',  
      'ILSVRC2012_val_00000262.JPEG',  
      'ILSVRC2012_val_00000293.JPEG',  
      'ILSVRC2012_val_00000307.JPEG',  
      'ILSVRC2012_val_00000329.JPEG',  
      'ILSVRC2012_val_00000414.JPEG',  
      'ILSVRC2012_val_00000439.JPEG',  
      'ILSVRC2012_val_00000476.JPEG',  
      'ILSVRC2012_val_00000570.JPEG',  
      'ILSVRC2012_val_00000638.JPEG',  
      'ILSVRC2012_val_00000651.JPEG',  
      'ILSVRC2012_val_00000713.JPEG',  
      'ILSVRC2012_val_00000747.JPEG',  
      'ILSVRC2012_val_00000837.JPEG',  
      'ILSVRC2012_val_00000873.JPEG',  
      'ILSVRC2012_val_00000880.JPEG',  
      'ILSVRC2012_val_00000911.JPEG',  
      'ILSVRC2012_val_00000921.JPEG',  
      'ILSVRC2012_val_00000962.JPEG',  
      'ILSVRC2012_val_00000963.JPEG',  
      'ILSVRC2012_val_00000994.JPEG',  
      'ILSVRC2012_val_00001024.JPEG',  
      'ILSVRC2012_val_00001031.JPEG',  
      'ILSVRC2012_val_00001084.JPEG',  
      'ILSVRC2012_val_00001114.JPEG',  
      'ILSVRC2012_val_00001274.JPEG',  
      'ILSVRC2012_val_00001368.JPEG',  
      'ILSVRC2012_val_00001398.JPEG',  
      'ILSVRC2012_val_00001462.JPEG',  
      'ILSVRC2012_val_00001508.JPEG',  
      'ILSVRC2012_val_00001556.JPEG',  
      'ILSVRC2012_val_00001598.JPEG',  
      'ILSVRC2012_val_00001676.JPEG',  
      'ILSVRC2012_val_00001720.JPEG',  
      'ILSVRC2012_val_00001751.JPEG',  
      'ILSVRC2012_val_00001906.JPEG',  
      'ILSVRC2012_val_00001915.JPEG',  
      'ILSVRC2012_val_00001935.JPEG',  
      'ILSVRC2012_val_00002023.JPEG',  
      'ILSVRC2012_val_00002052.JPEG',
```

'ILSVRC2012_val_00002071.JPEG',
'ILSVRC2012_val_00002138.JPEG',
'ILSVRC2012_val_00002191.JPEG',
'ILSVRC2012_val_00002207.JPEG',
'ILSVRC2012_val_00002241.JPEG',
'ILSVRC2012_val_00002284.JPEG',
'ILSVRC2012_val_00002338.JPEG',
'ILSVRC2012_val_00002357.JPEG',
'ILSVRC2012_val_00002372.JPEG',
'ILSVRC2012_val_00002425.JPEG',
'ILSVRC2012_val_00002449.JPEG',
'ILSVRC2012_val_00002480.JPEG',
'ILSVRC2012_val_00002536.JPEG',
'ILSVRC2012_val_00002655.JPEG',
'ILSVRC2012_val_00002668.JPEG',
'ILSVRC2012_val_00002681.JPEG',
'ILSVRC2012_val_00002730.JPEG',
'ILSVRC2012_val_00002752.JPEG',
'ILSVRC2012_val_00002825.JPEG',
'ILSVRC2012_val_00002848.JPEG',
'ILSVRC2012_val_00002872.JPEG',
'ILSVRC2012_val_00002922.JPEG',
'ILSVRC2012_val_00002929.JPEG',
'ILSVRC2012_val_00002969.JPEG',
'ILSVRC2012_val_00003014.JPEG',
'ILSVRC2012_val_00003092.JPEG',
'ILSVRC2012_val_00003116.JPEG',
'ILSVRC2012_val_00003150.JPEG',
'ILSVRC2012_val_00003183.JPEG',
'ILSVRC2012_val_00003204.JPEG',
'ILSVRC2012_val_00003279.JPEG',
'ILSVRC2012_val_00003297.JPEG',
'ILSVRC2012_val_00003386.JPEG',
'ILSVRC2012_val_00003518.JPEG',
'ILSVRC2012_val_00003548.JPEG',
'ILSVRC2012_val_00003557.JPEG',
'ILSVRC2012_val_00003558.JPEG',
'ILSVRC2012_val_00003605.JPEG',
'ILSVRC2012_val_00003606.JPEG',
'ILSVRC2012_val_00003665.JPEG',
'ILSVRC2012_val_00003735.JPEG',
'ILSVRC2012_val_00003762.JPEG',
'ILSVRC2012_val_00003816.JPEG',
'ILSVRC2012_val_00003861.JPEG',
'ILSVRC2012_val_00003878.JPEG',
'ILSVRC2012_val_00003903.JPEG',
'ILSVRC2012_val_00003936.JPEG',

'ILSVRC2012_val_00004006.JPEG',
'ILSVRC2012_val_00004037.JPEG',
'ILSVRC2012_val_00004183.JPEG',
'ILSVRC2012_val_00004243.JPEG',
'ILSVRC2012_val_00004245.JPEG',
'ILSVRC2012_val_00004306.JPEG',
'ILSVRC2012_val_00004311.JPEG',
'ILSVRC2012_val_00004315.JPEG',
'ILSVRC2012_val_00004316.JPEG',
'ILSVRC2012_val_00004317.JPEG',
'ILSVRC2012_val_00004329.JPEG',
'ILSVRC2012_val_00004344.JPEG',
'ILSVRC2012_val_00004375.JPEG',
'ILSVRC2012_val_00004417.JPEG',
'ILSVRC2012_val_00004463.JPEG',
'ILSVRC2012_val_00004518.JPEG',
'ILSVRC2012_val_00004550.JPEG',
'ILSVRC2012_val_00004559.JPEG',
'ILSVRC2012_val_00004577.JPEG',
'ILSVRC2012_val_00004655.JPEG',
'ILSVRC2012_val_00004677.JPEG',
'ILSVRC2012_val_00004698.JPEG',
'ILSVRC2012_val_00004747.JPEG',
'ILSVRC2012_val_00004749.JPEG',
'ILSVRC2012_val_00004756.JPEG',
'ILSVRC2012_val_00004873.JPEG',
'ILSVRC2012_val_00004893.JPEG',
'ILSVRC2012_val_00004969.JPEG',
'ILSVRC2012_val_00004983.JPEG',
'ILSVRC2012_val_00005156.JPEG',
'ILSVRC2012_val_00005224.JPEG',
'ILSVRC2012_val_00005287.JPEG',
'ILSVRC2012_val_00005336.JPEG',
'ILSVRC2012_val_00005339.JPEG',
'ILSVRC2012_val_00005412.JPEG',
'ILSVRC2012_val_00005427.JPEG',
'ILSVRC2012_val_00005432.JPEG',
'ILSVRC2012_val_00005492.JPEG',
'ILSVRC2012_val_00005505.JPEG',
'ILSVRC2012_val_00005535.JPEG',
'ILSVRC2012_val_00005567.JPEG',
'ILSVRC2012_val_00005576.JPEG',
'ILSVRC2012_val_00005594.JPEG',
'ILSVRC2012_val_00005642.JPEG',
'ILSVRC2012_val_00005833.JPEG',
'ILSVRC2012_val_00005848.JPEG',
'ILSVRC2012_val_00005870.JPEG',

'ILSVRC2012_val_00005931.JPEG',
'ILSVRC2012_val_00006007.JPEG',
'ILSVRC2012_val_00006030.JPEG',
'ILSVRC2012_val_00006171.JPEG',
'ILSVRC2012_val_00006195.JPEG',
'ILSVRC2012_val_00006216.JPEG',
'ILSVRC2012_val_00006251.JPEG',
'ILSVRC2012_val_00006267.JPEG',
'ILSVRC2012_val_00006321.JPEG',
'ILSVRC2012_val_00006326.JPEG',
'ILSVRC2012_val_00006328.JPEG',
'ILSVRC2012_val_00006402.JPEG',
'ILSVRC2012_val_00006463.JPEG',
'ILSVRC2012_val_00006491.JPEG',
'ILSVRC2012_val_00006526.JPEG',
'ILSVRC2012_val_00006554.JPEG',
'ILSVRC2012_val_00006597.JPEG',
'ILSVRC2012_val_00006603.JPEG',
'ILSVRC2012_val_00006697.JPEG',
'ILSVRC2012_val_00006763.JPEG',
'ILSVRC2012_val_00006835.JPEG',
'ILSVRC2012_val_00006851.JPEG',
'ILSVRC2012_val_00006907.JPEG',
'ILSVRC2012_val_00007129.JPEG',
'ILSVRC2012_val_00007197.JPEG',
'ILSVRC2012_val_00007231.JPEG',
'ILSVRC2012_val_00007237.JPEG',
'ILSVRC2012_val_00007263.JPEG',
'ILSVRC2012_val_00007405.JPEG',
'ILSVRC2012_val_00007530.JPEG',
'ILSVRC2012_val_00007569.JPEG',
'ILSVRC2012_val_00007667.JPEG',
'ILSVRC2012_val_00007729.JPEG',
'ILSVRC2012_val_00007743.JPEG',
'ILSVRC2012_val_00007853.JPEG',
'ILSVRC2012_val_00007872.JPEG',
'ILSVRC2012_val_00007875.JPEG',
'ILSVRC2012_val_00007912.JPEG',
'ILSVRC2012_val_00008024.JPEG',
'ILSVRC2012_val_00008108.JPEG',
'ILSVRC2012_val_00008125.JPEG',
'ILSVRC2012_val_00008176.JPEG',
'ILSVRC2012_val_00008294.JPEG',
'ILSVRC2012_val_00008347.JPEG',
'ILSVRC2012_val_00008420.JPEG',
'ILSVRC2012_val_00008495.JPEG',
'ILSVRC2012_val_00008510.JPEG',

'ILSVRC2012_val_00008512.JPEG',
'ILSVRC2012_val_00008640.JPEG',
'ILSVRC2012_val_00008749.JPEG',
'ILSVRC2012_val_00008786.JPEG',
'ILSVRC2012_val_00008816.JPEG',
'ILSVRC2012_val_00008848.JPEG',
'ILSVRC2012_val_00008898.JPEG',
'ILSVRC2012_val_00008952.JPEG',
'ILSVRC2012_val_00008963.JPEG',
'ILSVRC2012_val_00009034.JPEG',
'ILSVRC2012_val_00009092.JPEG',
'ILSVRC2012_val_00009111.JPEG',
'ILSVRC2012_val_00009191.JPEG',
'ILSVRC2012_val_00009208.JPEG',
'ILSVRC2012_val_00009219.JPEG',
'ILSVRC2012_val_00009237.JPEG',
'ILSVRC2012_val_00009239.JPEG',
'ILSVRC2012_val_00009346.JPEG',
'ILSVRC2012_val_00009351.JPEG',
'ILSVRC2012_val_00009374.JPEG',
'ILSVRC2012_val_00009379.JPEG',
'ILSVRC2012_val_00009396.JPEG',
'ILSVRC2012_val_00009401.JPEG',
'ILSVRC2012_val_00009420.JPEG',
'ILSVRC2012_val_00009460.JPEG',
'ILSVRC2012_val_00009510.JPEG',
'ILSVRC2012_val_00009518.JPEG',
'ILSVRC2012_val_00009535.JPEG',
'ILSVRC2012_val_00009597.JPEG',
'ILSVRC2012_val_00009615.JPEG',
'ILSVRC2012_val_00009663.JPEG',
'ILSVRC2012_val_00009734.JPEG',
'ILSVRC2012_val_00009807.JPEG',
'ILSVRC2012_val_00009831.JPEG',
'ILSVRC2012_val_00009978.JPEG',
'ILSVRC2012_val_00009981.JPEG',
'ILSVRC2012_val_00010074.JPEG',
'ILSVRC2012_val_00010128.JPEG',
'ILSVRC2012_val_00010162.JPEG',
'ILSVRC2012_val_00010208.JPEG',
'ILSVRC2012_val_00010240.JPEG',
'ILSVRC2012_val_00010272.JPEG',
'ILSVRC2012_val_00010306.JPEG',
'ILSVRC2012_val_00010310.JPEG',
'ILSVRC2012_val_00010339.JPEG',
'ILSVRC2012_val_00010352.JPEG',
'ILSVRC2012_val_00010363.JPEG',

'ILSVRC2012_val_00010393.JPEG',
'ILSVRC2012_val_00010409.JPEG',
'ILSVRC2012_val_00010443.JPEG',
'ILSVRC2012_val_00010456.JPEG',
'ILSVRC2012_val_00010501.JPEG',
'ILSVRC2012_val_00010509.JPEG',
'ILSVRC2012_val_00010594.JPEG',
'ILSVRC2012_val_00010689.JPEG',
'ILSVRC2012_val_00010763.JPEG',
'ILSVRC2012_val_00010767.JPEG',
'ILSVRC2012_val_00010861.JPEG',
'ILSVRC2012_val_00010880.JPEG',
'ILSVRC2012_val_00010999.JPEG',
'ILSVRC2012_val_00011056.JPEG',
'ILSVRC2012_val_00011171.JPEG',
'ILSVRC2012_val_00011178.JPEG',
'ILSVRC2012_val_00011233.JPEG',
'ILSVRC2012_val_00011249.JPEG',
'ILSVRC2012_val_00011267.JPEG',
'ILSVRC2012_val_00011331.JPEG',
'ILSVRC2012_val_00011388.JPEG',
'ILSVRC2012_val_00011403.JPEG',
'ILSVRC2012_val_00011420.JPEG',
'ILSVRC2012_val_00011446.JPEG',
'ILSVRC2012_val_00011501.JPEG',
'ILSVRC2012_val_00011519.JPEG',
'ILSVRC2012_val_00011646.JPEG',
'ILSVRC2012_val_00011782.JPEG',
'ILSVRC2012_val_00011841.JPEG',
'ILSVRC2012_val_00011914.JPEG',
'ILSVRC2012_val_00011915.JPEG',
'ILSVRC2012_val_00011993.JPEG',
'ILSVRC2012_val_00012058.JPEG',
'ILSVRC2012_val_00012132.JPEG',
'ILSVRC2012_val_00012259.JPEG',
'ILSVRC2012_val_00012328.JPEG',
'ILSVRC2012_val_00012484.JPEG',
'ILSVRC2012_val_00012503.JPEG',
'ILSVRC2012_val_00012528.JPEG',
'ILSVRC2012_val_00012546.JPEG',
'ILSVRC2012_val_00012651.JPEG',
'ILSVRC2012_val_00012797.JPEG',
'ILSVRC2012_val_00012814.JPEG',
'ILSVRC2012_val_00012836.JPEG',
'ILSVRC2012_val_00012870.JPEG',
'ILSVRC2012_val_00012879.JPEG',
'ILSVRC2012_val_00012880.JPEG',

'ILSVRC2012_val_00012939.JPEG',
'ILSVRC2012_val_00013209.JPEG',
'ILSVRC2012_val_00013223.JPEG',
'ILSVRC2012_val_00013270.JPEG',
'ILSVRC2012_val_00013277.JPEG',
'ILSVRC2012_val_00013321.JPEG',
'ILSVRC2012_val_00013367.JPEG',
'ILSVRC2012_val_00013407.JPEG',
'ILSVRC2012_val_00013456.JPEG',
'ILSVRC2012_val_00013484.JPEG',
'ILSVRC2012_val_00013504.JPEG',
'ILSVRC2012_val_00013512.JPEG',
'ILSVRC2012_val_00013513.JPEG',
'ILSVRC2012_val_00013623.JPEG',
'ILSVRC2012_val_00013628.JPEG',
'ILSVRC2012_val_00013655.JPEG',
'ILSVRC2012_val_00013716.JPEG',
'ILSVRC2012_val_00013753.JPEG',
'ILSVRC2012_val_00013847.JPEG',
'ILSVRC2012_val_00013902.JPEG',
'ILSVRC2012_val_00013945.JPEG',
'ILSVRC2012_val_00013955.JPEG',
'ILSVRC2012_val_00013998.JPEG',
'ILSVRC2012_val_00014046.JPEG',
'ILSVRC2012_val_00014090.JPEG',
'ILSVRC2012_val_00014108.JPEG',
'ILSVRC2012_val_00014139.JPEG',
'ILSVRC2012_val_00014180.JPEG',
'ILSVRC2012_val_00014328.JPEG',
'ILSVRC2012_val_00014336.JPEG',
'ILSVRC2012_val_00014362.JPEG',
'ILSVRC2012_val_00014467.JPEG',
'ILSVRC2012_val_00014673.JPEG',
'ILSVRC2012_val_00014678.JPEG',
'ILSVRC2012_val_00014741.JPEG',
'ILSVRC2012_val_00014771.JPEG',
'ILSVRC2012_val_00014879.JPEG',
'ILSVRC2012_val_00014917.JPEG',
'ILSVRC2012_val_00015045.JPEG',
'ILSVRC2012_val_00015139.JPEG',
'ILSVRC2012_val_00015157.JPEG',
'ILSVRC2012_val_00015195.JPEG',
'ILSVRC2012_val_00015216.JPEG',
'ILSVRC2012_val_00015220.JPEG',
'ILSVRC2012_val_00015223.JPEG',
'ILSVRC2012_val_00015242.JPEG',
'ILSVRC2012_val_00015264.JPEG',

'ILSVRC2012_val_00015266.JPEG',
'ILSVRC2012_val_00015269.JPEG',
'ILSVRC2012_val_00015373.JPEG',
'ILSVRC2012_val_00015410.JPEG',
'ILSVRC2012_val_00015416.JPEG',
'ILSVRC2012_val_00015431.JPEG',
'ILSVRC2012_val_00015545.JPEG',
'ILSVRC2012_val_00015549.JPEG',
'ILSVRC2012_val_00015555.JPEG',
'ILSVRC2012_val_00015631.JPEG',
'ILSVRC2012_val_00015706.JPEG',
'ILSVRC2012_val_00015732.JPEG',
'ILSVRC2012_val_00015830.JPEG',
'ILSVRC2012_val_00015863.JPEG',
'ILSVRC2012_val_00015903.JPEG',
'ILSVRC2012_val_00015922.JPEG',
'ILSVRC2012_val_00015964.JPEG',
'ILSVRC2012_val_00016018.JPEG',
'ILSVRC2012_val_00016052.JPEG',
'ILSVRC2012_val_00016168.JPEG',
'ILSVRC2012_val_00016198.JPEG',
'ILSVRC2012_val_00016252.JPEG',
'ILSVRC2012_val_00016259.JPEG',
'ILSVRC2012_val_00016331.JPEG',
'ILSVRC2012_val_00016347.JPEG',
'ILSVRC2012_val_00016395.JPEG',
'ILSVRC2012_val_00016476.JPEG',
'ILSVRC2012_val_00016538.JPEG',
'ILSVRC2012_val_00016548.JPEG',
'ILSVRC2012_val_00016677.JPEG',
'ILSVRC2012_val_00016781.JPEG',
'ILSVRC2012_val_00016794.JPEG',
'ILSVRC2012_val_00016831.JPEG',
'ILSVRC2012_val_00016834.JPEG',
'ILSVRC2012_val_00016840.JPEG',
'ILSVRC2012_val_00016859.JPEG',
'ILSVRC2012_val_00016909.JPEG',
'ILSVRC2012_val_00016962.JPEG',
'ILSVRC2012_val_00016988.JPEG',
'ILSVRC2012_val_00017017.JPEG',
'ILSVRC2012_val_00017024.JPEG',
'ILSVRC2012_val_00017028.JPEG',
'ILSVRC2012_val_00017105.JPEG',
'ILSVRC2012_val_00017194.JPEG',
'ILSVRC2012_val_00017196.JPEG',
'ILSVRC2012_val_00017310.JPEG',
'ILSVRC2012_val_00017384.JPEG',

'ILSVRC2012_val_00017391.JPEG',
'ILSVRC2012_val_00017440.JPEG',
'ILSVRC2012_val_00017472.JPEG',
'ILSVRC2012_val_00017577.JPEG',
'ILSVRC2012_val_00017648.JPEG',
'ILSVRC2012_val_00017660.JPEG',
'ILSVRC2012_val_00017699.JPEG',
'ILSVRC2012_val_00017700.JPEG',
'ILSVRC2012_val_00017712.JPEG',
'ILSVRC2012_val_00017756.JPEG',
'ILSVRC2012_val_00017839.JPEG',
'ILSVRC2012_val_00017916.JPEG',
'ILSVRC2012_val_00017970.JPEG',
'ILSVRC2012_val_00017975.JPEG',
'ILSVRC2012_val_00017976.JPEG',
'ILSVRC2012_val_00017995.JPEG',
'ILSVRC2012_val_00018075.JPEG',
'ILSVRC2012_val_00018145.JPEG',
'ILSVRC2012_val_00018162.JPEG',
'ILSVRC2012_val_00018167.JPEG',
'ILSVRC2012_val_00018183.JPEG',
'ILSVRC2012_val_00018250.JPEG',
'ILSVRC2012_val_00018280.JPEG',
'ILSVRC2012_val_00018317.JPEG',
'ILSVRC2012_val_00018466.JPEG',
'ILSVRC2012_val_00018672.JPEG',
'ILSVRC2012_val_00018676.JPEG',
'ILSVRC2012_val_00018680.JPEG',
'ILSVRC2012_val_00018692.JPEG',
'ILSVRC2012_val_00018699.JPEG',
'ILSVRC2012_val_00018943.JPEG',
'ILSVRC2012_val_00018948.JPEG',
'ILSVRC2012_val_00019034.JPEG',
'ILSVRC2012_val_00019038.JPEG',
'ILSVRC2012_val_00019082.JPEG',
'ILSVRC2012_val_00019101.JPEG',
'ILSVRC2012_val_00019184.JPEG',
'ILSVRC2012_val_00019208.JPEG',
'ILSVRC2012_val_00019236.JPEG',
'ILSVRC2012_val_00019409.JPEG',
'ILSVRC2012_val_00019423.JPEG',
'ILSVRC2012_val_00019424.JPEG',
'ILSVRC2012_val_00019435.JPEG',
'ILSVRC2012_val_00019459.JPEG',
'ILSVRC2012_val_00019531.JPEG',
'ILSVRC2012_val_00019721.JPEG',
'ILSVRC2012_val_00019744.JPEG',

'ILSVRC2012_val_00019788.JPEG',
'ILSVRC2012_val_00019988.JPEG',
'ILSVRC2012_val_00020011.JPEG',
'ILSVRC2012_val_00020106.JPEG',
'ILSVRC2012_val_00020178.JPEG',
'ILSVRC2012_val_00020349.JPEG',
'ILSVRC2012_val_00020418.JPEG',
'ILSVRC2012_val_00020436.JPEG',
'ILSVRC2012_val_00020451.JPEG',
'ILSVRC2012_val_00020469.JPEG',
'ILSVRC2012_val_00020495.JPEG',
'ILSVRC2012_val_00020610.JPEG',
'ILSVRC2012_val_00020660.JPEG',
'ILSVRC2012_val_00020730.JPEG',
'ILSVRC2012_val_00020785.JPEG',
'ILSVRC2012_val_00020788.JPEG',
'ILSVRC2012_val_00020822.JPEG',
'ILSVRC2012_val_00020889.JPEG',
'ILSVRC2012_val_00020905.JPEG',
'ILSVRC2012_val_00021085.JPEG',
'ILSVRC2012_val_00021239.JPEG',
'ILSVRC2012_val_00021300.JPEG',
'ILSVRC2012_val_00021325.JPEG',
'ILSVRC2012_val_00021430.JPEG',
'ILSVRC2012_val_00021461.JPEG',
'ILSVRC2012_val_00021523.JPEG',
'ILSVRC2012_val_00021650.JPEG',
'ILSVRC2012_val_00021656.JPEG',
'ILSVRC2012_val_00021740.JPEG',
'ILSVRC2012_val_00021793.JPEG',
'ILSVRC2012_val_00021905.JPEG',
'ILSVRC2012_val_00021956.JPEG',
'ILSVRC2012_val_00022025.JPEG',
'ILSVRC2012_val_00022076.JPEG',
'ILSVRC2012_val_00022106.JPEG',
'ILSVRC2012_val_00022138.JPEG',
'ILSVRC2012_val_00022186.JPEG',
'ILSVRC2012_val_00022248.JPEG',
'ILSVRC2012_val_00022387.JPEG',
'ILSVRC2012_val_00022407.JPEG',
'ILSVRC2012_val_00022505.JPEG',
'ILSVRC2012_val_00022552.JPEG',
'ILSVRC2012_val_00022644.JPEG',
'ILSVRC2012_val_00022727.JPEG',
'ILSVRC2012_val_00022865.JPEG',
'ILSVRC2012_val_00022889.JPEG',
'ILSVRC2012_val_00022972.JPEG',

'ILSVRC2012_val_00023074.JPEG',
'ILSVRC2012_val_00023125.JPEG',
'ILSVRC2012_val_00023162.JPEG',
'ILSVRC2012_val_00023185.JPEG',
'ILSVRC2012_val_00023187.JPEG',
'ILSVRC2012_val_00023214.JPEG',
'ILSVRC2012_val_00023239.JPEG',
'ILSVRC2012_val_00023438.JPEG',
'ILSVRC2012_val_00023441.JPEG',
'ILSVRC2012_val_00023443.JPEG',
'ILSVRC2012_val_00023551.JPEG',
'ILSVRC2012_val_00023559.JPEG',
'ILSVRC2012_val_00023573.JPEG',
'ILSVRC2012_val_00023607.JPEG',
'ILSVRC2012_val_00023698.JPEG',
'ILSVRC2012_val_00023766.JPEG',
'ILSVRC2012_val_00023778.JPEG',
'ILSVRC2012_val_00023850.JPEG',
'ILSVRC2012_val_00023869.JPEG',
'ILSVRC2012_val_00023895.JPEG',
'ILSVRC2012_val_00023907.JPEG',
'ILSVRC2012_val_00023916.JPEG',
'ILSVRC2012_val_00023935.JPEG',
'ILSVRC2012_val_00023942.JPEG',
'ILSVRC2012_val_00023945.JPEG',
'ILSVRC2012_val_00023986.JPEG',
'ILSVRC2012_val_00024012.JPEG',
'ILSVRC2012_val_00024015.JPEG',
'ILSVRC2012_val_00024025.JPEG',
'ILSVRC2012_val_00024044.JPEG',
'ILSVRC2012_val_00024059.JPEG',
'ILSVRC2012_val_00024204.JPEG',
'ILSVRC2012_val_00024235.JPEG',
'ILSVRC2012_val_00024256.JPEG',
'ILSVRC2012_val_00024268.JPEG',
'ILSVRC2012_val_00024327.JPEG',
'ILSVRC2012_val_00024353.JPEG',
'ILSVRC2012_val_00024643.JPEG',
'ILSVRC2012_val_00024704.JPEG',
'ILSVRC2012_val_00024719.JPEG',
'ILSVRC2012_val_00024800.JPEG',
'ILSVRC2012_val_00024854.JPEG',
'ILSVRC2012_val_00024884.JPEG',
'ILSVRC2012_val_00024895.JPEG',
'ILSVRC2012_val_00024940.JPEG',
'ILSVRC2012_val_00024957.JPEG',
'ILSVRC2012_val_00025035.JPEG',

'ILSVRC2012_val_00025037.JPEG',
'ILSVRC2012_val_00025051.JPEG',
'ILSVRC2012_val_00025129.JPEG',
'ILSVRC2012_val_00025133.JPEG',
'ILSVRC2012_val_00025137.JPEG',
'ILSVRC2012_val_00025151.JPEG',
'ILSVRC2012_val_00025214.JPEG',
'ILSVRC2012_val_00025324.JPEG',
'ILSVRC2012_val_00025347.JPEG',
'ILSVRC2012_val_00025361.JPEG',
'ILSVRC2012_val_00025428.JPEG',
'ILSVRC2012_val_00025431.JPEG',
'ILSVRC2012_val_00025476.JPEG',
'ILSVRC2012_val_00025488.JPEG',
'ILSVRC2012_val_00025507.JPEG',
'ILSVRC2012_val_00025524.JPEG',
'ILSVRC2012_val_00025526.JPEG',
'ILSVRC2012_val_00025527.JPEG',
'ILSVRC2012_val_00025535.JPEG',
'ILSVRC2012_val_00025577.JPEG',
'ILSVRC2012_val_00025627.JPEG',
'ILSVRC2012_val_00025640.JPEG',
'ILSVRC2012_val_00025686.JPEG',
'ILSVRC2012_val_00025692.JPEG',
'ILSVRC2012_val_00025806.JPEG',
'ILSVRC2012_val_00025888.JPEG',
'ILSVRC2012_val_00025900.JPEG',
'ILSVRC2012_val_00025969.JPEG',
'ILSVRC2012_val_00026001.JPEG',
'ILSVRC2012_val_00026010.JPEG',
'ILSVRC2012_val_00026023.JPEG',
'ILSVRC2012_val_00026064.JPEG',
'ILSVRC2012_val_00026123.JPEG',
'ILSVRC2012_val_00026136.JPEG',
'ILSVRC2012_val_00026259.JPEG',
'ILSVRC2012_val_00026271.JPEG',
'ILSVRC2012_val_00026279.JPEG',
'ILSVRC2012_val_00026362.JPEG',
'ILSVRC2012_val_00026397.JPEG',
'ILSVRC2012_val_00026465.JPEG',
'ILSVRC2012_val_00026490.JPEG',
'ILSVRC2012_val_00026508.JPEG',
'ILSVRC2012_val_00026541.JPEG',
'ILSVRC2012_val_00026656.JPEG',
'ILSVRC2012_val_00026673.JPEG',
'ILSVRC2012_val_00026687.JPEG',
'ILSVRC2012_val_00026755.JPEG',

'ILSVRC2012_val_00026776.JPEG',
'ILSVRC2012_val_00026784.JPEG',
'ILSVRC2012_val_00026908.JPEG',
'ILSVRC2012_val_00026929.JPEG',
'ILSVRC2012_val_00027005.JPEG',
'ILSVRC2012_val_00027063.JPEG',
'ILSVRC2012_val_00027113.JPEG',
'ILSVRC2012_val_00027145.JPEG',
'ILSVRC2012_val_00027160.JPEG',
'ILSVRC2012_val_00027207.JPEG',
'ILSVRC2012_val_00027254.JPEG',
'ILSVRC2012_val_00027283.JPEG',
'ILSVRC2012_val_00027301.JPEG',
'ILSVRC2012_val_00027351.JPEG',
'ILSVRC2012_val_00027374.JPEG',
'ILSVRC2012_val_00027415.JPEG',
'ILSVRC2012_val_00027419.JPEG',
'ILSVRC2012_val_00027498.JPEG',
'ILSVRC2012_val_00027507.JPEG',
'ILSVRC2012_val_00027514.JPEG',
'ILSVRC2012_val_00027535.JPEG',
'ILSVRC2012_val_00027619.JPEG',
'ILSVRC2012_val_00027633.JPEG',
'ILSVRC2012_val_00027762.JPEG',
'ILSVRC2012_val_00027811.JPEG',
'ILSVRC2012_val_00027831.JPEG',
'ILSVRC2012_val_00028024.JPEG',
'ILSVRC2012_val_00028028.JPEG',
'ILSVRC2012_val_00028063.JPEG',
'ILSVRC2012_val_00028084.JPEG',
'ILSVRC2012_val_00028099.JPEG',
'ILSVRC2012_val_00028107.JPEG',
'ILSVRC2012_val_00028158.JPEG',
'ILSVRC2012_val_00028161.JPEG',
'ILSVRC2012_val_00028181.JPEG',
'ILSVRC2012_val_00028221.JPEG',
'ILSVRC2012_val_00028224.JPEG',
'ILSVRC2012_val_00028230.JPEG',
'ILSVRC2012_val_00028422.JPEG',
'ILSVRC2012_val_00028440.JPEG',
'ILSVRC2012_val_00028450.JPEG',
'ILSVRC2012_val_00028524.JPEG',
'ILSVRC2012_val_00028564.JPEG',
'ILSVRC2012_val_00028607.JPEG',
'ILSVRC2012_val_00028712.JPEG',
'ILSVRC2012_val_00028713.JPEG',
'ILSVRC2012_val_00028791.JPEG',

'ILSVRC2012_val_00028902.JPEG',
'ILSVRC2012_val_00028978.JPEG',
'ILSVRC2012_val_00029169.JPEG',
'ILSVRC2012_val_00029300.JPEG',
'ILSVRC2012_val_00029310.JPEG',
'ILSVRC2012_val_00029417.JPEG',
'ILSVRC2012_val_00029430.JPEG',
'ILSVRC2012_val_00029441.JPEG',
'ILSVRC2012_val_00029481.JPEG',
'ILSVRC2012_val_00029495.JPEG',
'ILSVRC2012_val_00029568.JPEG',
'ILSVRC2012_val_00029581.JPEG',
'ILSVRC2012_val_00029619.JPEG',
'ILSVRC2012_val_00029627.JPEG',
'ILSVRC2012_val_00029695.JPEG',
'ILSVRC2012_val_00029709.JPEG',
'ILSVRC2012_val_00029740.JPEG',
'ILSVRC2012_val_00029835.JPEG',
'ILSVRC2012_val_00029886.JPEG',
'ILSVRC2012_val_00029921.JPEG',
'ILSVRC2012_val_00029930.JPEG',
'ILSVRC2012_val_00029999.JPEG',
'ILSVRC2012_val_00030011.JPEG',
'ILSVRC2012_val_00030017.JPEG',
'ILSVRC2012_val_00030042.JPEG',
'ILSVRC2012_val_00030060.JPEG',
'ILSVRC2012_val_00030071.JPEG',
'ILSVRC2012_val_00030073.JPEG',
'ILSVRC2012_val_00030127.JPEG',
'ILSVRC2012_val_00030156.JPEG',
'ILSVRC2012_val_00030217.JPEG',
'ILSVRC2012_val_00030224.JPEG',
'ILSVRC2012_val_00030283.JPEG',
'ILSVRC2012_val_00030320.JPEG',
'ILSVRC2012_val_00030321.JPEG',
'ILSVRC2012_val_00030367.JPEG',
'ILSVRC2012_val_00030380.JPEG',
'ILSVRC2012_val_00030470.JPEG',
'ILSVRC2012_val_00030475.JPEG',
'ILSVRC2012_val_00030544.JPEG',
'ILSVRC2012_val_00030657.JPEG',
'ILSVRC2012_val_00030658.JPEG',
'ILSVRC2012_val_00030699.JPEG',
'ILSVRC2012_val_00030701.JPEG',
'ILSVRC2012_val_00030740.JPEG',
'ILSVRC2012_val_00030845.JPEG',
'ILSVRC2012_val_00030911.JPEG',

'ILSVRC2012_val_00030926.JPEG',
'ILSVRC2012_val_00030966.JPEG',
'ILSVRC2012_val_00031071.JPEG',
'ILSVRC2012_val_00031086.JPEG',
'ILSVRC2012_val_00031094.JPEG',
'ILSVRC2012_val_00031138.JPEG',
'ILSVRC2012_val_00031169.JPEG',
'ILSVRC2012_val_00031171.JPEG',
'ILSVRC2012_val_00031239.JPEG',
'ILSVRC2012_val_00031243.JPEG',
'ILSVRC2012_val_00031273.JPEG',
'ILSVRC2012_val_00031333.JPEG',
'ILSVRC2012_val_00031424.JPEG',
'ILSVRC2012_val_00031466.JPEG',
'ILSVRC2012_val_00031481.JPEG',
'ILSVRC2012_val_00031512.JPEG',
'ILSVRC2012_val_00031568.JPEG',
'ILSVRC2012_val_00031630.JPEG',
'ILSVRC2012_val_00031649.JPEG',
'ILSVRC2012_val_00031716.JPEG',
'ILSVRC2012_val_00031720.JPEG',
'ILSVRC2012_val_00031772.JPEG',
'ILSVRC2012_val_00031808.JPEG',
'ILSVRC2012_val_00031913.JPEG',
'ILSVRC2012_val_00031973.JPEG',
'ILSVRC2012_val_00031987.JPEG',
'ILSVRC2012_val_00032025.JPEG',
'ILSVRC2012_val_00032102.JPEG',
'ILSVRC2012_val_00032130.JPEG',
'ILSVRC2012_val_00032154.JPEG',
'ILSVRC2012_val_00032162.JPEG',
'ILSVRC2012_val_00032164.JPEG',
'ILSVRC2012_val_00032169.JPEG',
'ILSVRC2012_val_00032235.JPEG',
'ILSVRC2012_val_00032258.JPEG',
'ILSVRC2012_val_00032288.JPEG',
'ILSVRC2012_val_00032310.JPEG',
'ILSVRC2012_val_00032359.JPEG',
'ILSVRC2012_val_00032394.JPEG',
'ILSVRC2012_val_00032417.JPEG',
'ILSVRC2012_val_00032441.JPEG',
'ILSVRC2012_val_00032586.JPEG',
'ILSVRC2012_val_00032667.JPEG',
'ILSVRC2012_val_00032675.JPEG',
'ILSVRC2012_val_00032702.JPEG',
'ILSVRC2012_val_00032710.JPEG',
'ILSVRC2012_val_00032788.JPEG',

'ILSVRC2012_val_00032831.JPEG',
'ILSVRC2012_val_00032849.JPEG',
'ILSVRC2012_val_00032855.JPEG',
'ILSVRC2012_val_00032930.JPEG',
'ILSVRC2012_val_00032993.JPEG',
'ILSVRC2012_val_00033070.JPEG',
'ILSVRC2012_val_00033092.JPEG',
'ILSVRC2012_val_00033150.JPEG',
'ILSVRC2012_val_00033181.JPEG',
'ILSVRC2012_val_00033196.JPEG',
'ILSVRC2012_val_00033307.JPEG',
'ILSVRC2012_val_00033318.JPEG',
'ILSVRC2012_val_00033337.JPEG',
'ILSVRC2012_val_00033372.JPEG',
'ILSVRC2012_val_00033483.JPEG',
'ILSVRC2012_val_00033515.JPEG',
'ILSVRC2012_val_00033538.JPEG',
'ILSVRC2012_val_00033580.JPEG',
'ILSVRC2012_val_00033691.JPEG',
'ILSVRC2012_val_00033770.JPEG',
'ILSVRC2012_val_00033942.JPEG',
'ILSVRC2012_val_00034018.JPEG',
'ILSVRC2012_val_00034038.JPEG',
'ILSVRC2012_val_00034095.JPEG',
'ILSVRC2012_val_00034115.JPEG',
'ILSVRC2012_val_00034144.JPEG',
'ILSVRC2012_val_00034178.JPEG',
'ILSVRC2012_val_00034242.JPEG',
'ILSVRC2012_val_00034250.JPEG',
'ILSVRC2012_val_00034316.JPEG',
'ILSVRC2012_val_00034333.JPEG',
'ILSVRC2012_val_00034386.JPEG',
'ILSVRC2012_val_00034413.JPEG',
'ILSVRC2012_val_00034437.JPEG',
'ILSVRC2012_val_00034482.JPEG',
'ILSVRC2012_val_00034552.JPEG',
'ILSVRC2012_val_00034654.JPEG',
'ILSVRC2012_val_00034720.JPEG',
'ILSVRC2012_val_00034837.JPEG',
'ILSVRC2012_val_00034864.JPEG',
'ILSVRC2012_val_00034929.JPEG',
'ILSVRC2012_val_00034950.JPEG',
'ILSVRC2012_val_00034971.JPEG',
'ILSVRC2012_val_00035060.JPEG',
'ILSVRC2012_val_00035104.JPEG',
'ILSVRC2012_val_00035123.JPEG',
'ILSVRC2012_val_00035131.JPEG',

'ILSVRC2012_val_00035141.JPEG',
'ILSVRC2012_val_00035151.JPEG',
'ILSVRC2012_val_00035223.JPEG',
'ILSVRC2012_val_00035254.JPEG',
'ILSVRC2012_val_00035303.JPEG',
'ILSVRC2012_val_00035348.JPEG',
'ILSVRC2012_val_00035367.JPEG',
'ILSVRC2012_val_00035389.JPEG',
'ILSVRC2012_val_00035405.JPEG',
'ILSVRC2012_val_00035495.JPEG',
'ILSVRC2012_val_00035540.JPEG',
'ILSVRC2012_val_00035565.JPEG',
'ILSVRC2012_val_00035735.JPEG',
'ILSVRC2012_val_00035774.JPEG',
'ILSVRC2012_val_00035785.JPEG',
'ILSVRC2012_val_00035807.JPEG',
'ILSVRC2012_val_00035833.JPEG',
'ILSVRC2012_val_00035872.JPEG',
'ILSVRC2012_val_00035947.JPEG',
'ILSVRC2012_val_00036008.JPEG',
'ILSVRC2012_val_00036015.JPEG',
'ILSVRC2012_val_00036042.JPEG',
'ILSVRC2012_val_00036105.JPEG',
'ILSVRC2012_val_00036118.JPEG',
'ILSVRC2012_val_00036157.JPEG',
'ILSVRC2012_val_00036284.JPEG',
'ILSVRC2012_val_00036368.JPEG',
'ILSVRC2012_val_00036420.JPEG',
'ILSVRC2012_val_00036431.JPEG',
'ILSVRC2012_val_00036479.JPEG',
'ILSVRC2012_val_00036534.JPEG',
'ILSVRC2012_val_00036737.JPEG',
'ILSVRC2012_val_00036749.JPEG',
'ILSVRC2012_val_00036812.JPEG',
'ILSVRC2012_val_00036819.JPEG',
'ILSVRC2012_val_00036999.JPEG',
'ILSVRC2012_val_00037013.JPEG',
'ILSVRC2012_val_00037050.JPEG',
'ILSVRC2012_val_00037070.JPEG',
'ILSVRC2012_val_00037116.JPEG',
'ILSVRC2012_val_00037135.JPEG',
'ILSVRC2012_val_00037239.JPEG',
'ILSVRC2012_val_00037260.JPEG',
'ILSVRC2012_val_00037328.JPEG',
'ILSVRC2012_val_00037375.JPEG',
'ILSVRC2012_val_00037383.JPEG',
'ILSVRC2012_val_00037420.JPEG',

'ILSVRC2012_val_00037436.JPEG',
'ILSVRC2012_val_00037454.JPEG',
'ILSVRC2012_val_00037488.JPEG',
'ILSVRC2012_val_00037516.JPEG',
'ILSVRC2012_val_00037569.JPEG',
'ILSVRC2012_val_00037581.JPEG',
'ILSVRC2012_val_00037596.JPEG',
'ILSVRC2012_val_00037611.JPEG',
'ILSVRC2012_val_00037698.JPEG',
'ILSVRC2012_val_00037705.JPEG',
'ILSVRC2012_val_00037719.JPEG',
'ILSVRC2012_val_00037740.JPEG',
'ILSVRC2012_val_00037748.JPEG',
'ILSVRC2012_val_00037753.JPEG',
'ILSVRC2012_val_00037780.JPEG',
'ILSVRC2012_val_00037798.JPEG',
'ILSVRC2012_val_00037806.JPEG',
'ILSVRC2012_val_00037824.JPEG',
'ILSVRC2012_val_00037834.JPEG',
'ILSVRC2012_val_00037846.JPEG',
'ILSVRC2012_val_00037861.JPEG',
'ILSVRC2012_val_00037915.JPEG',
'ILSVRC2012_val_00037920.JPEG',
'ILSVRC2012_val_00037939.JPEG',
'ILSVRC2012_val_00037941.JPEG',
'ILSVRC2012_val_00037998.JPEG',
'ILSVRC2012_val_00038057.JPEG',
'ILSVRC2012_val_00038091.JPEG',
'ILSVRC2012_val_00038156.JPEG',
'ILSVRC2012_val_00038159.JPEG',
'ILSVRC2012_val_00038160.JPEG',
'ILSVRC2012_val_00038192.JPEG',
'ILSVRC2012_val_00038215.JPEG',
'ILSVRC2012_val_00038372.JPEG',
'ILSVRC2012_val_00038405.JPEG',
'ILSVRC2012_val_00038488.JPEG',
'ILSVRC2012_val_00038557.JPEG',
'ILSVRC2012_val_00038794.JPEG',
'ILSVRC2012_val_00038826.JPEG',
'ILSVRC2012_val_00038895.JPEG',
'ILSVRC2012_val_00038935.JPEG',
'ILSVRC2012_val_00039004.JPEG',
'ILSVRC2012_val_00039067.JPEG',
'ILSVRC2012_val_00039219.JPEG',
'ILSVRC2012_val_00039304.JPEG',
'ILSVRC2012_val_00039308.JPEG',
'ILSVRC2012_val_00039355.JPEG',

'ILSVRC2012_val_00039445.JPEG',
'ILSVRC2012_val_00039466.JPEG',
'ILSVRC2012_val_00039525.JPEG',
'ILSVRC2012_val_00039534.JPEG',
'ILSVRC2012_val_00039571.JPEG',
'ILSVRC2012_val_00039588.JPEG',
'ILSVRC2012_val_00039648.JPEG',
'ILSVRC2012_val_00039759.JPEG',
'ILSVRC2012_val_00039801.JPEG',
'ILSVRC2012_val_00039814.JPEG',
'ILSVRC2012_val_00039865.JPEG',
'ILSVRC2012_val_00039905.JPEG',
'ILSVRC2012_val_00039913.JPEG',
'ILSVRC2012_val_00039943.JPEG',
'ILSVRC2012_val_00040038.JPEG',
'ILSVRC2012_val_00040071.JPEG',
'ILSVRC2012_val_00040098.JPEG',
'ILSVRC2012_val_00040342.JPEG',
'ILSVRC2012_val_00040358.JPEG',
'ILSVRC2012_val_00040388.JPEG',
'ILSVRC2012_val_00040412.JPEG',
'ILSVRC2012_val_00040426.JPEG',
'ILSVRC2012_val_00040439.JPEG',
'ILSVRC2012_val_00040538.JPEG',
'ILSVRC2012_val_00040567.JPEG',
'ILSVRC2012_val_00040606.JPEG',
'ILSVRC2012_val_00040710.JPEG',
'ILSVRC2012_val_00040783.JPEG',
'ILSVRC2012_val_00040833.JPEG',
'ILSVRC2012_val_00040843.JPEG',
'ILSVRC2012_val_00040903.JPEG',
'ILSVRC2012_val_00040938.JPEG',
'ILSVRC2012_val_00041051.JPEG',
'ILSVRC2012_val_00041114.JPEG',
'ILSVRC2012_val_00041135.JPEG',
'ILSVRC2012_val_00041152.JPEG',
'ILSVRC2012_val_00041168.JPEG',
'ILSVRC2012_val_00041206.JPEG',
'ILSVRC2012_val_00041248.JPEG',
'ILSVRC2012_val_00041252.JPEG',
'ILSVRC2012_val_00041308.JPEG',
'ILSVRC2012_val_00041334.JPEG',
'ILSVRC2012_val_00041400.JPEG',
'ILSVRC2012_val_00041595.JPEG',
'ILSVRC2012_val_00041635.JPEG',
'ILSVRC2012_val_00041836.JPEG',
'ILSVRC2012_val_00041852.JPEG',

'ILSVRC2012_val_00041872.JPEG',
'ILSVRC2012_val_00041939.JPEG',
'ILSVRC2012_val_00041944.JPEG',
'ILSVRC2012_val_00041977.JPEG',
'ILSVRC2012_val_00041990.JPEG',
'ILSVRC2012_val_00042032.JPEG',
'ILSVRC2012_val_00042048.JPEG',
'ILSVRC2012_val_00042135.JPEG',
'ILSVRC2012_val_00042175.JPEG',
'ILSVRC2012_val_00042245.JPEG',
'ILSVRC2012_val_00042246.JPEG',
'ILSVRC2012_val_00042248.JPEG',
'ILSVRC2012_val_00042345.JPEG',
'ILSVRC2012_val_00042370.JPEG',
'ILSVRC2012_val_00042385.JPEG',
'ILSVRC2012_val_00042506.JPEG',
'ILSVRC2012_val_00042534.JPEG',
'ILSVRC2012_val_00042547.JPEG',
'ILSVRC2012_val_00042568.JPEG',
'ILSVRC2012_val_00042600.JPEG',
'ILSVRC2012_val_00042751.JPEG',
'ILSVRC2012_val_00042926.JPEG',
'ILSVRC2012_val_00043229.JPEG',
'ILSVRC2012_val_00043300.JPEG',
'ILSVRC2012_val_00043323.JPEG',
'ILSVRC2012_val_00043387.JPEG',
'ILSVRC2012_val_00043392.JPEG',
'ILSVRC2012_val_00043566.JPEG',
'ILSVRC2012_val_00043585.JPEG',
'ILSVRC2012_val_00043702.JPEG',
'ILSVRC2012_val_00043751.JPEG',
'ILSVRC2012_val_00043764.JPEG',
'ILSVRC2012_val_00043788.JPEG',
'ILSVRC2012_val_00043965.JPEG',
'ILSVRC2012_val_00043995.JPEG',
'ILSVRC2012_val_00044009.JPEG',
'ILSVRC2012_val_00044038.JPEG',
'ILSVRC2012_val_00044052.JPEG',
'ILSVRC2012_val_00044095.JPEG',
'ILSVRC2012_val_00044192.JPEG',
'ILSVRC2012_val_00044276.JPEG',
'ILSVRC2012_val_00044311.JPEG',
'ILSVRC2012_val_00044378.JPEG',
'ILSVRC2012_val_00044483.JPEG',
'ILSVRC2012_val_00044507.JPEG',
'ILSVRC2012_val_00044540.JPEG',
'ILSVRC2012_val_00044624.JPEG',

'ILSVRC2012_val_00044657.JPEG',
'ILSVRC2012_val_00044706.JPEG',
'ILSVRC2012_val_00044788.JPEG',
'ILSVRC2012_val_00044896.JPEG',
'ILSVRC2012_val_00044975.JPEG',
'ILSVRC2012_val_00045062.JPEG',
'ILSVRC2012_val_00045107.JPEG',
'ILSVRC2012_val_00045133.JPEG',
'ILSVRC2012_val_00045150.JPEG',
'ILSVRC2012_val_00045172.JPEG',
'ILSVRC2012_val_00045198.JPEG',
'ILSVRC2012_val_00045294.JPEG',
'ILSVRC2012_val_00045389.JPEG',
'ILSVRC2012_val_00045402.JPEG',
'ILSVRC2012_val_00045415.JPEG',
'ILSVRC2012_val_00045480.JPEG',
'ILSVRC2012_val_00045502.JPEG',
'ILSVRC2012_val_00045547.JPEG',
'ILSVRC2012_val_00045551.JPEG',
'ILSVRC2012_val_00045568.JPEG',
'ILSVRC2012_val_00045580.JPEG',
'ILSVRC2012_val_00045581.JPEG',
'ILSVRC2012_val_00045611.JPEG',
'ILSVRC2012_val_00045623.JPEG',
'ILSVRC2012_val_00045730.JPEG',
'ILSVRC2012_val_00045761.JPEG',
'ILSVRC2012_val_00045785.JPEG',
'ILSVRC2012_val_00045866.JPEG',
'ILSVRC2012_val_00045873.JPEG',
'ILSVRC2012_val_00045880.JPEG',
'ILSVRC2012_val_00045925.JPEG',
'ILSVRC2012_val_00045936.JPEG',
'ILSVRC2012_val_00045948.JPEG',
'ILSVRC2012_val_00046071.JPEG',
'ILSVRC2012_val_00046159.JPEG',
'ILSVRC2012_val_00046171.JPEG',
'ILSVRC2012_val_00046173.JPEG',
'ILSVRC2012_val_00046252.JPEG',
'ILSVRC2012_val_00046260.JPEG',
'ILSVRC2012_val_00046280.JPEG',
'ILSVRC2012_val_00046393.JPEG',
'ILSVRC2012_val_00046415.JPEG',
'ILSVRC2012_val_00046422.JPEG',
'ILSVRC2012_val_00046497.JPEG',
'ILSVRC2012_val_00046499.JPEG',
'ILSVRC2012_val_00046524.JPEG',
'ILSVRC2012_val_00046654.JPEG',

'ILSVRC2012_val_00046751.JPEG',
'ILSVRC2012_val_00046797.JPEG',
'ILSVRC2012_val_00046833.JPEG',
'ILSVRC2012_val_00046915.JPEG',
'ILSVRC2012_val_00046917.JPEG',
'ILSVRC2012_val_00046969.JPEG',
'ILSVRC2012_val_00047029.JPEG',
'ILSVRC2012_val_00047080.JPEG',
'ILSVRC2012_val_00047121.JPEG',
'ILSVRC2012_val_00047137.JPEG',
'ILSVRC2012_val_00047272.JPEG',
'ILSVRC2012_val_00047320.JPEG',
'ILSVRC2012_val_00047326.JPEG',
'ILSVRC2012_val_00047333.JPEG',
'ILSVRC2012_val_00047396.JPEG',
'ILSVRC2012_val_00047400.JPEG',
'ILSVRC2012_val_00047431.JPEG',
...]