

法律声明

本课件包括演示文稿、示例、代码、题库、视频和声音等内容，深度之眼和讲师拥有完全知识产权；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或者机构不得盗版、复制、仿造其中的创意和内容，我们保留一切通过法律手段追究违反者的权利。

课程详情请咨询

- 微信公众号：深度之眼
- 客服微信号：deepshare0920



公众号



微信

关注公众号深度之眼，后台回复论文，获取60篇AI必读经典前沿论文



deepshare.net

深度之眼

transforms图像增强 (二)

导师：余老师

关注公众号深度之眼，后台回复论文，获取60篇AI必读经典前沿论文

目录

1/transforms —— 图像变换

2/transforms —— transforms方法操作

3/自定义transforms方法

图像变换

Data Augmentation

transforms

transforms



1. Pad

功能：对图片边缘进行填充

- **padding**：设置填充大小

当为a时，上下左右均填充a个像素

当为(a, b)时，上下填充b个像素，左右填充a个像素

当为(a, b, c, d)时，左，上，右，下分别填充a, b, c, d

- **padding_mode**：填充模式，有4种模式，constant、edge、reflect和symmetric
- **fill**：constant时，设置填充的像素值，(R, G, B) or (Gray)

```
transforms.Pad(padding,  
                fill=0,  
                padding_mode='constant')
```

transforms

transforms

2. ColorJitter

功能：调整亮度、对比度、饱和度和色相

- **brightness**：亮度调整因子

当为a时，从 $[\max(0, 1-a), 1+a]$ 中随机选择

当为(a, b)时，从[a, b]中

- **contrast**：对比度参数，同brightness
- **saturation**：饱和度参数，同brightness
- **hue**：色相参数，当为a时，从 $[-a, a]$ 中选择参数，注： $0 \leq a \leq 0.5$

当为(a, b)时，从[a, b]中选择参数，注： $-0.5 \leq a \leq b \leq 0.5$

```
transforms.ColorJitter(brightness=0,  
                        contrast=0,  
                        saturation=0,  
                        hue=0)
```

transforms

transforms



3. Grayscale

4. RandomGrayscale

功能：依概率将图片转换为灰度图

- **num_output_channels**: 输出通道数
只能设1或3
- **p**: 概率值，图像被转换为灰度图的概率

```
RandomGrayscale(num_output_channels,  
                 p=0.1)
```

```
Grayscale(num_output_channels)
```

transforms

transforms



5. RandomAffine

功能：对图像进行仿射变换，仿射变换是二维的线性变换，由五种基本原子变换构成，分别是**旋转、平移、缩放、错切和翻转**

```
RandomAffine(degrees,  
              translate=None,  
              scale=None,  
              shear=None,  
              resample=False,  
              fillcolor=0)
```

- **degrees**：旋转角度设置
- **translate**：平移区间设置，如(a, b), a设置宽 (width) , b设置高(height)

图像在宽维度平移的区间为 $-img_width * a < dx < img_width * a$

- **scale**：缩放比例（以面积为单位）
- **fill_color**：填充颜色设置

关注公众号深度之眼，后台回复论文，获取60篇AI必读经典前沿论文

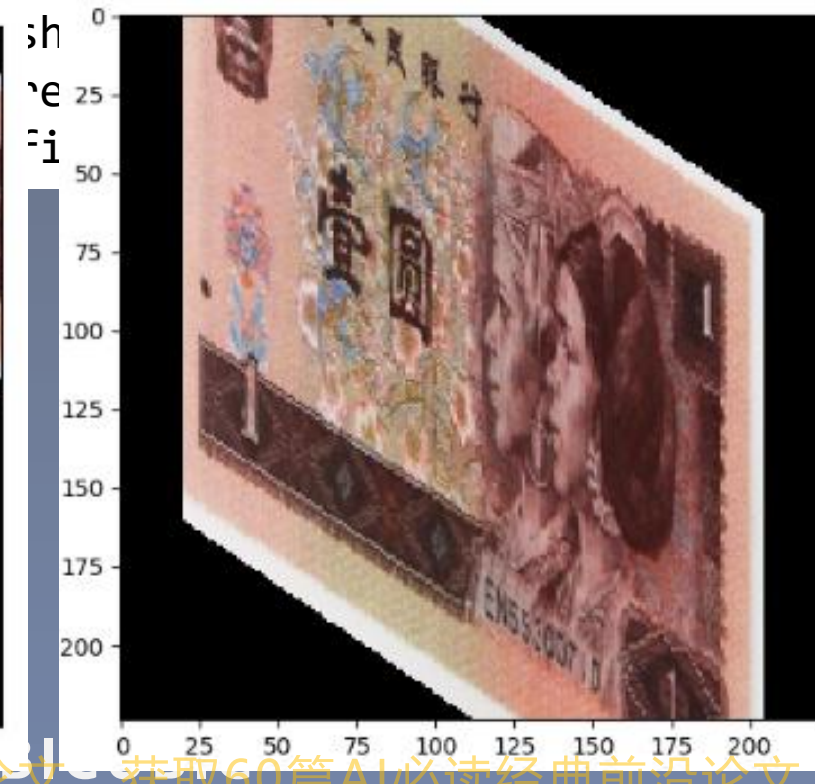
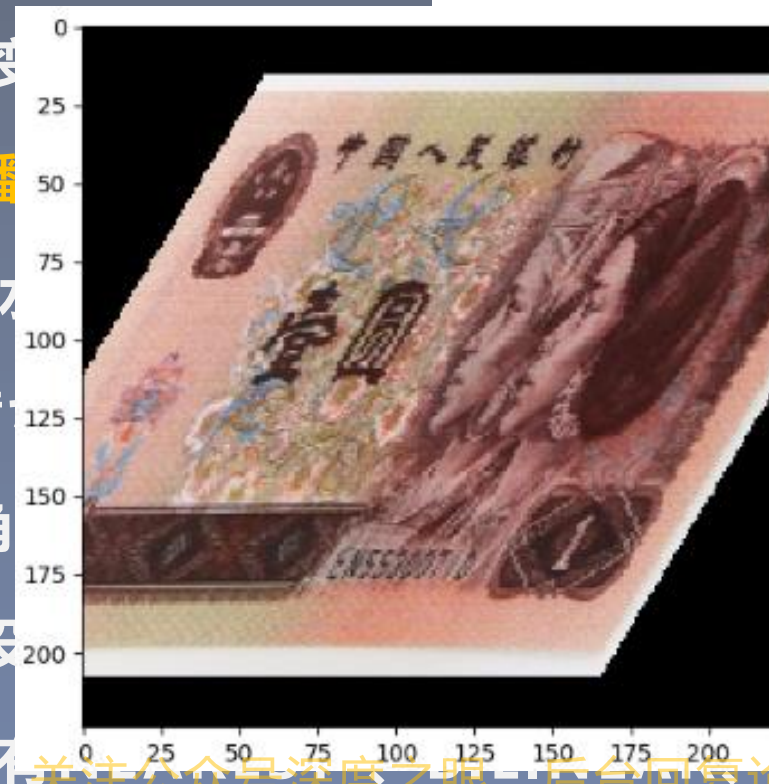
transforms

transforms

5. RandomAffine

功能：对图像进行仿射变换，仿射变换是二维

`RandomAffine`(degrees,
translate=None,
scale=None,



transforms

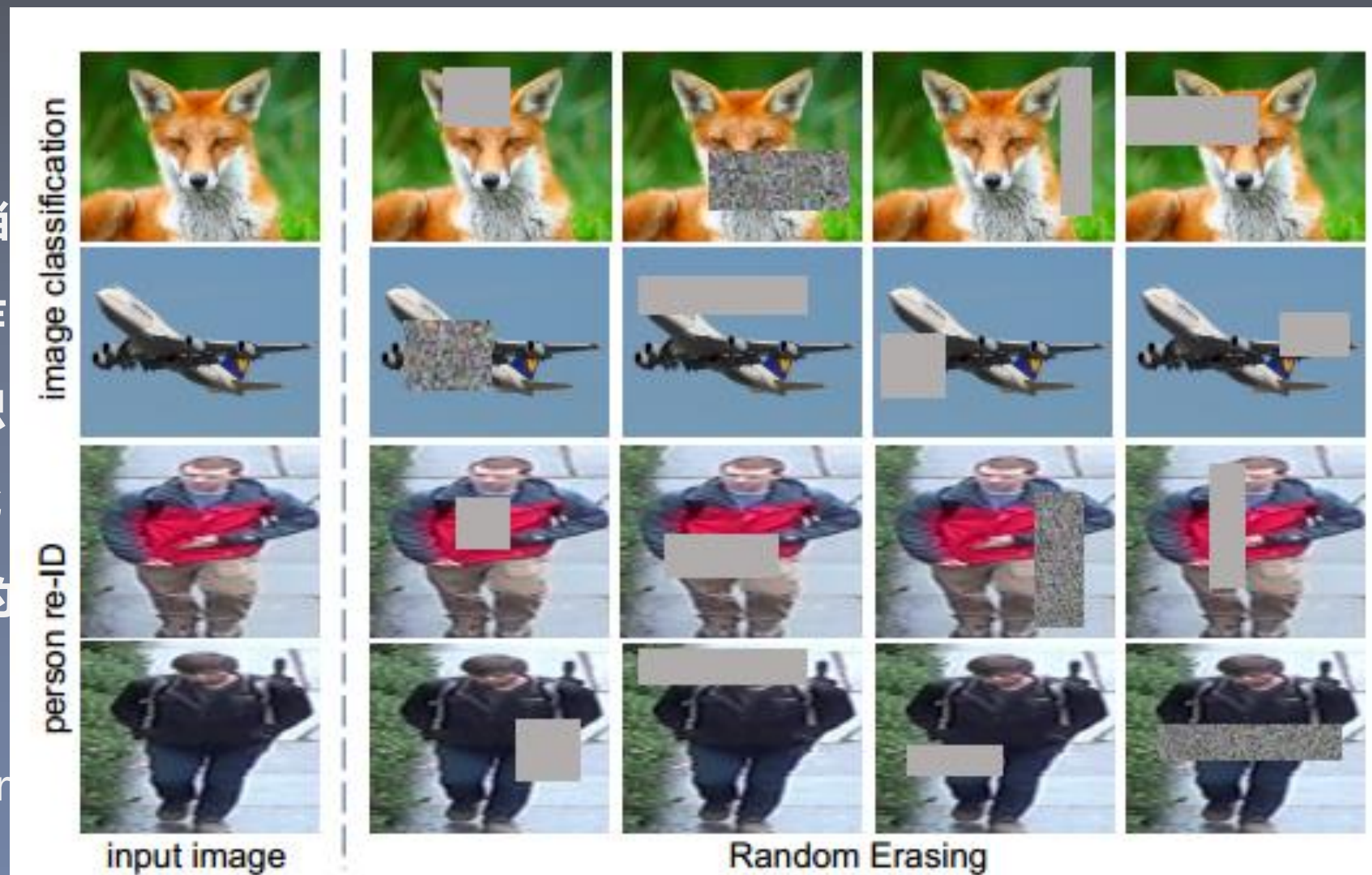
transforms

6. RandomErasing

功能：对图像进行随机遮挡

- **p**：概率值，执行该操作
- **scale**：遮挡区域的面积
- **ratio**：遮挡区域长宽比
- **value**：设置遮挡区域的

参考文献：《Random Erasing



transforms

transforms



7. transforms.Lambda

功能：用户自定义lambda方法

- **lambda**: lambda匿名函数

lambda [arg1 [,arg2, ... , argn]] : expression

eg:

```
transforms.TenCrop(200, vertical_flip=True),
```

```
transforms.Lambda(lambda crops: torch.stack([transforms.ToTensor()(crop) for crop in crops])),
```

```
transforms.Lambda(lambd)
```


transforms的操作

Transforms Operation

transforms的操作

Transforms Operation

1. transforms.RandomChoice

功能：从一系列transforms方法中随机挑选一个

```
transforms.RandomChoice([transforms1, transforms2, transforms3])
```

2. transforms.RandomApply

功能：依据概率执行一组transforms操作

```
transforms.RandomApply([transforms1, transforms2, transforms3], p=0.5)
```

3. transforms.RandomOrder

功能：对一组transforms操作打乱顺序

```
transforms.RandomOrder([transforms1, transforms2, transforms3])
```


自定义transforms

User-Defined Transforms

自定义transforms

User-Defined Transforms



自定义transforms要素：

1. 仅接收一个参数，返回一个参数
2. 注意上下游的输出与输入

通过类实现多参数传入：

```
class YourTransforms(object):  
    def __init__(self, ...):  
        ...  
    def __call__(self, img):  
        ...  
        return img
```

```
class Compose(object):  
    def __call__(self, img):  
        for t in self.transforms:  
            img = t(img)  
        return img
```

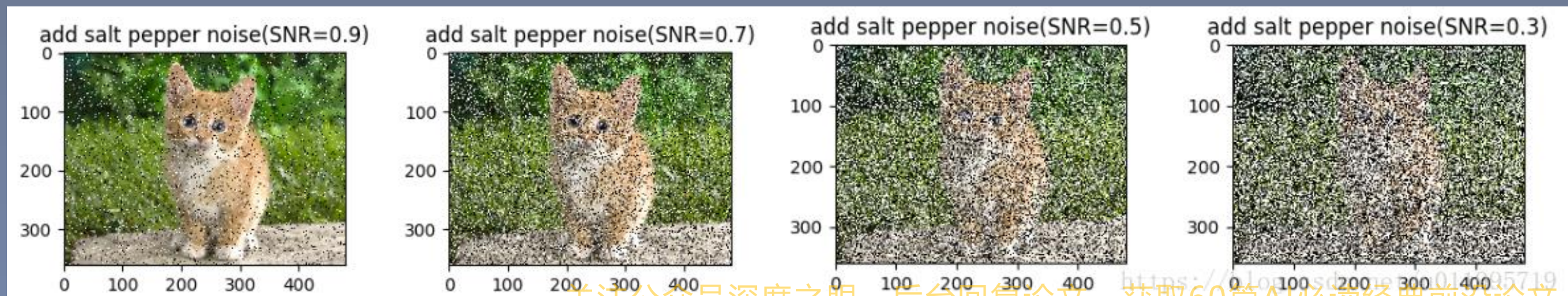
自定义transforms

User-Defined Transforms

椒盐噪声

椒盐噪声又称为脉冲噪声，是一种随机出现的白点或者黑点，白点称为盐噪声，黑色为椒噪声

信噪比 (Signal-Noise Rate, SNR) 是衡量噪声的比例，图像中为图像像素的占比



自定义transforms

User-Defined Transforms



椒盐噪声

```
class AddPepperNoise(object):  
  
    def __init__(self, snr, p):  
        self.snr = snr  
        self.p = p  
  
    def __call__(self, img):  
        ...  
        添加椒盐噪声具体实现过程  
        ...  
  
    return img
```

```
class Compose(object):  
    def __call__(self, img):  
        for t in self.transforms:  
            img = t(img)  
        return img
```

transforms方法



Transforms Methods

一、裁剪

- 1. transforms.CenterCrop
- 2. transforms.RandomCrop
- 3. transforms.RandomResizedCrop
- 4. transforms.FiveCrop
- 5. transforms.TenCrop

二、翻转和旋转

- 1. transforms.RandomHorizontalFlip
- 2. transforms.RandomVerticalFlip
- 3. transforms.RandomRotation

三、图像变换

- 1. transforms.Pad
- 2. transforms.ColorJitter

- 3. transforms.Grayscale
- 4. transforms.RandomGrayscale
- 5. transforms.RandomAffine
- 6. transforms.LinearTransformation
- 7. transforms.RandomErasing
- 8. transforms.Lambda
- 9. transforms.Resize
- 10. transforms.ToTensor
- 11. transforms.Normalize

四、transforms的操作

- 1. transforms.RandomChoice
- 2. transforms.RandomApply

- 3. transforms.RandomOrder
- 关注公众号深度之眼，后台回复论文，获取60篇AI必读经典前沿论文

数据增强实战应用

Data Augmentation Application

数据增强实战

原则：让训练集与测试集更接近

- **空间位置**：平移
- **色彩**：灰度图，色彩抖动
- **形状**：仿射变换
- **上下文场景**：遮挡，填充
-



训练集



测试集

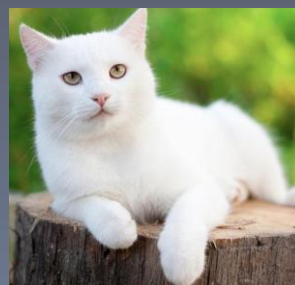
数据增强实战应用

Data Augmentation Application

数据增强实战

原则：让训练集与测试集更接近

- **空间位置**：平移
- **色彩**：灰度图，色彩抖动
- **形状**：仿射变换
- **上下文场景**：遮挡，填充
-



训练集



测试集

数据增强实战应用

Data Augmentation Application

人民币分类：

第四套RMB



第五套RMB



—— 结 语 ——

在这次课程中，学习了数据预处理transforms的
图像变换、操作方法以及自定义transforms

在下次课程中，我们将会学习PyTorch的

模型模块



关注公众号深度之眼，后台回复论文，获取60篇AI必读经典前沿论文



deepshare.net

深度之眼

联系我们:

电话: 18001992849

邮箱: service@deepshare.net

QQ: 2677693114



公众号



客服微信

关注公众号深度之眼，后台回复论文，获取60篇AI必读经典前沿论文