Image Augmentation

A반 3조 배혜연

Image augmentation : 이미지 변환을 통해 이미지를 늘리는 방법

Albumentations : **Albumentations : Fast and Flexible Image Augmentations** 라는 이름으로 2020년 Information 저널에 출판
다른 라이브러리들과 비교하였을 때 빠르다는 것이 가장 큰 특징이자 장점

	albumentations 0.4.5	imgaug 0.4.0	torchvision (Pillow-SIMD backend) 0.5.0	keras 2.3.1	augmentor 0.2.8	solt 0.1.8
HorizontalFlip	3066	1544	1652	874	1658	853
VerticalFlip	4159	2014	1427	4147	1448	3788
Rotate	417	327	160	29	60	113
ShiftScaleRotate	703	471	144	30	-	-
Brightness	2210	997	397	210	396	2058
Contrast	2208	1023	330	-	331	2059
BrightnessContrast	2199	582	190	-	190	1051
ShiftRGB	2215	998	-	378	-	-
ShiftHSV	381	241	59	-	-	128
Gamma	2340	-	686	-	-	951
Grayscale	4961	372	735	-	1423	4286
RandomCrop64	157376	2560	41448	-	36036	35454
PadToSize512	2833	-	478	-	-	2629
Resize512	952	595	885	-	873	881
RandomSizedCrop_64_512	3128	881	1295	-	1254	2678
Equalize	760	399	-	-	666	-
Multiply	2184	1059	-	-	-	-
MultiplyElementwise	124	197	-	-	-	-

> Single core에서 초당 처리되는 image 수

> ShiftScaleRotate

class **albumentations.augmentations.transforms.ShiftScaleRotate**(*shift_limit=0.0625, scale_limit=0.1, rotate_limit=45, interpolation=1, border_mode=4, always_apply=False, p=0.5*)   [source]

Randomly apply affine transforms: translate, scale and rotate the input.

입력변환, 크기 조정 및 회전

Parameters:

- **shift_limit** (*(float, float) or float*) – shift factor range for both height and width. If shift_limit is a single float value, the range will be (-shift_limit, shift_limit). Absolute values for lower and upper bounds should lie in range [0, 1]. Default: 0.0625.
- **scale_limit** (*(float, float) or float*) – scaling factor range. If scale_limit is a single float value, the range will be (-scale_limit, scale_limit). Default: 0.1.
- **rotate_limit** (*(int, int) or int*) – rotation range. If rotate_limit is a single int value, the range will be (-rotate_limit, rotate_limit). Default: 45.
- **interpolation** (*OpenCV flag*) – flag that is used to specify the interpolation algorithm. Should be one of: cv2.INTER_NEAREST, cv2.INTER_LINEAR, cv2.INTER_CUBIC, cv2.INTER_AREA, cv2.INTER_LANCZOS4. Default: cv2.INTER_LINEAR.
- **border_mode** (*OpenCV flag*) – flag that is used to specify the pixel extrapolation method. Should be one of: cv2.BORDER_CONSTANT, cv2.BORDER_REPLICATE, cv2.BORDER_REFLECT, cv2.BORDER_WRAP, cv2.BORDER_REFLECT_101. Default: cv2.BORDER_REFLECT_101
- **p** (*float*) – probability of applying the transform. Default: 0.5.

Targets:
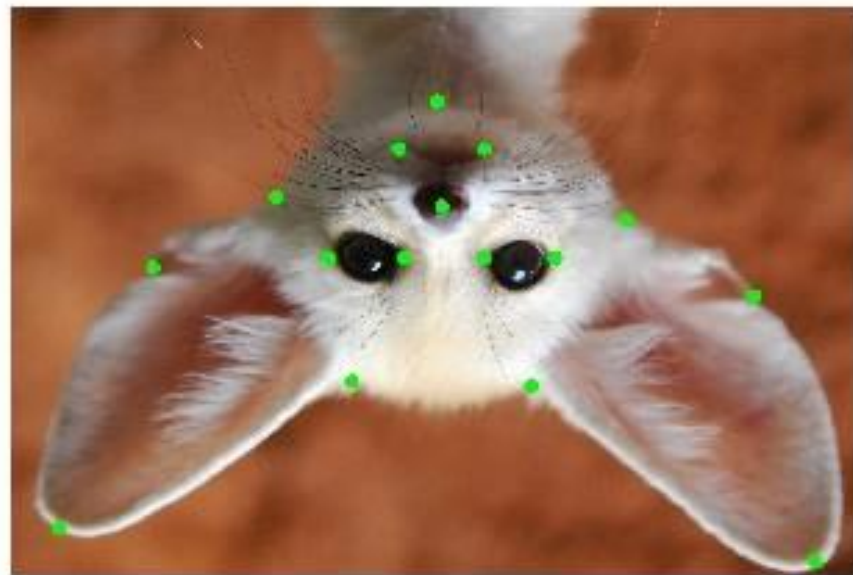
image, mask

Image types:

uint8, float32

>GridDistortion

격자로 나눠서 왜곡, 움직임, 굴곡, 외삽   *Extrapolation(외삽) : 범위 밖의 값을 예측한다

```
class albumentations.augmentations.transforms.GridDistortion(num_steps=5, distort_limit=0.3,
interpolation=1, border_mode=4, always_apply=False, p=0.5)    [source]
```

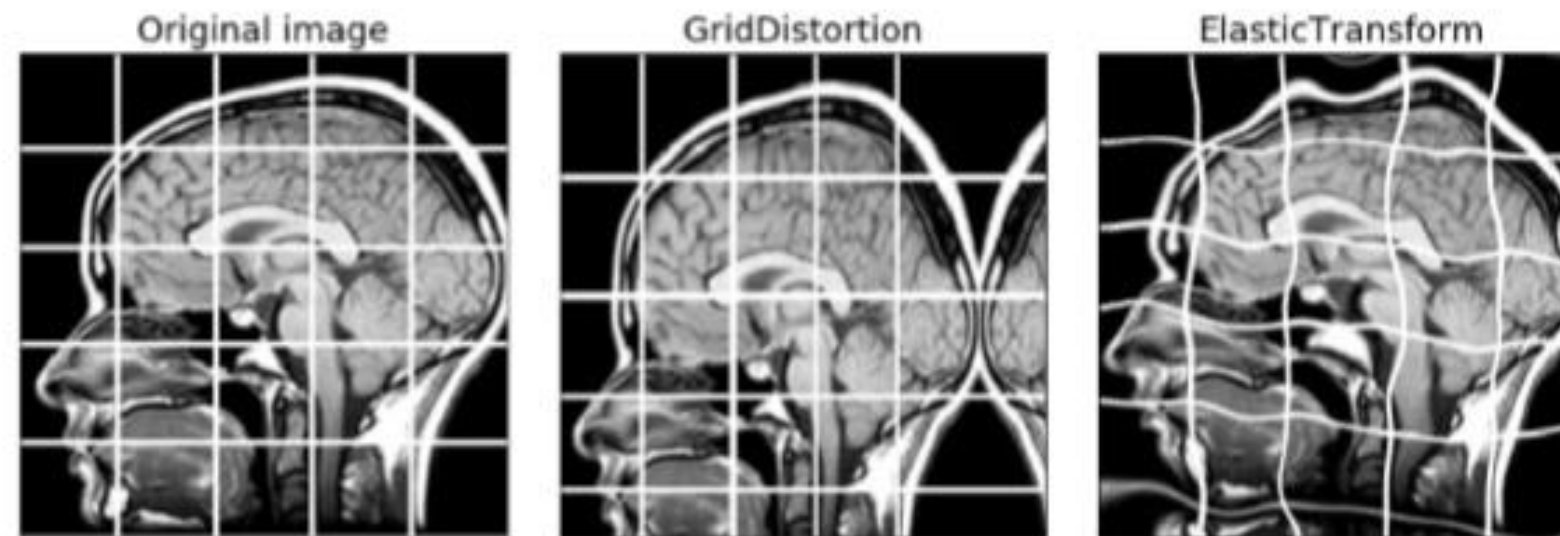Targets:

image, mask

Image types:

uint8, float32

Fig. 4. Grid distortion and elastic transform applied to a medical image.

>Blur

class albumentations.augmentations.transforms.Blur(*blur_limit=7, always_apply=False, p=0.5*)
    [source]

Blur the input image using a random-sized kernel.

Parameters:
- **blur_limit** (*int*) – maximum kernel size for blurring the input image. Default: 7.
- **p** (*float*) – probability of applying the transform. Default: 0.5.

Input image에 대해 블러 처리를 하기 위한 최대 커널 사이즈 (default=7)

변환 적용의 가능성 (default=0.5)

Targets:

    image

Image types:

    uint8, float32

Blur

1. Original image
2. `Blur(blur_limit=(7, 7), p=1)`
3. `Blur(blur_limit=(15, 15), p=1)`
4. `Blur(blur_limit=(50, 50), p=1)`
5. `Blur(blur_limit=(100, 100), p=1)`
6. `Blur(blur_limit=(300, 300), p=1)`

https://github.com/albumentations-team/albumentations_examples

감사합니다