

## تمرین 3 یادگیری عمیق

### توضیحات سوال 3

امیرعباس افضلی 400100662

## توضیحات دیتاست:

ابتدا توابع گفته شده را ایمپلمنت کردم و از توابع 4 تا 6 برای data augmentation بر روی داده های ترین استفاده شده است. توابع گفته شده را تا حد امکان سعی شده است که با torchvision ایمپلمنت شود که عملیات بر روی gpu انجام شود. فقط در توابع displacement و rotation به ناچار از CV2 استفاده شده است. (ایمپلمنتیشن سایر توابع که با CV2 انجام شده است، کامنت شده است). و عملکرد این توابع را برای یک مثال خاص، در نوت بوک آمده است.

و کلاس دیتاست نیز به نحوی پیاده سازی شده است که در تابع \_\_getitem\_\_ آن، لیبل هر دیتا، یک بردار سه تایی است که ممکن است 0، 1 و یا دو المان یک داشته باشد. که هر یک از 1 ها به این معنا است که operation متناظر بر روس تصویر انجام شده است. و عکس اصلی، عکس تغییر یافته و لیبل را خروجی می دهد. همچنین بدلیل حجم بالای دیتای ترینینگ، آن را کوچک کردیم و از یک subset ای از آن استفاده شده است.

## پیاده سازی شبکه

شبکه ی مورد نظر همانطور که خواست سوال بود، پیاده سازی شد.

### سوال تئوری ۱:

۱. مقادیر زیر را برای لایه های کانولوشن چه قدر در نظر گرفته اید؟

channel out size (conv(۱\*۱))

تعداد outchannel در هر بلاک به در هنگام ساخت هر بلاک در تابع init آن تعیین می شود. به طور کلی در init آبجکتی از block، in\_channels و out\_channels ورودی گرفته می شود و تعداد چنل های ورودی و خروجی کانولوشن اول، دوم و کانولوشن residual به ترتیب به شکل زیر است:

```
self.conv1 = nn.Conv2d(in_channels, out_channels, kernel_size=3,padding=1)
```

```
self.conv2 = nn.Conv2d(out_channels, out_channels, kernel_size=3,padding=1)
```

```
self.conv2_residual = nn.Conv2d(in_channels, out_channels, kernel_size=1)
```

padding(conv)

پدینگ هم در لایه های کانولوشنی اول و دوم، ۱ در نظر گرفته شده است. زیرا kernel size = 3 می باشد و برای اینکه shape خروجی و ورودی کانولوشن یکسان باشد (زیرا در ادامه از skip connection استفاده می کنیم)، پدینگ داده شده است.

۲. دلیل استفاده از کانولوشن ۱ در چیست؟

معمولا برای کنترل تعداد کانال ها از conv1x1 استفاده می شود. که تعداد channel های فیچر مپ خروجی از آن برابر تعداد فیلتر های 1x1 می باشد.

### سوال تئوری:

بررسی کنید عکس ها با یک، ۲ یا ۳ تغییر کدام بهتر تشخیص داده میشوند؟

عکس با سه تغییر که استفاده نشده است. ولی عکس هایی با دو تغییر سخت تر از عکس های با یک تغییر تشخیص داده می شوند.

```
number of images with 0 transformation which have recognized correctly:
484 out of 531 >>> 91.1487758945386%

number of images with 1 transformation which have recognized correctly:
847 out of 1863 >>> 45.46430488459474%

number of images with 2 transformation which have recognized correctly:
340 out of 908 >>> 37.44493392070485%
```

و با توجه به نتایج تشخیص جابه جایی از همه بهتر، سپس scaling و در آخر rotation می باشد.

## نتائج train & validation :

