



تمرین شماره دو

شبکه عصبی پیچشی (CNN)

درس یادگیری عمیق | بهار ۱۴۰۱

استاد درس: جناب آقای دکتر حامد ملک

گیت هاب درس:

<https://github.com/SBU-CE/Deep-Learning>

بخش اول؛ سوالات:

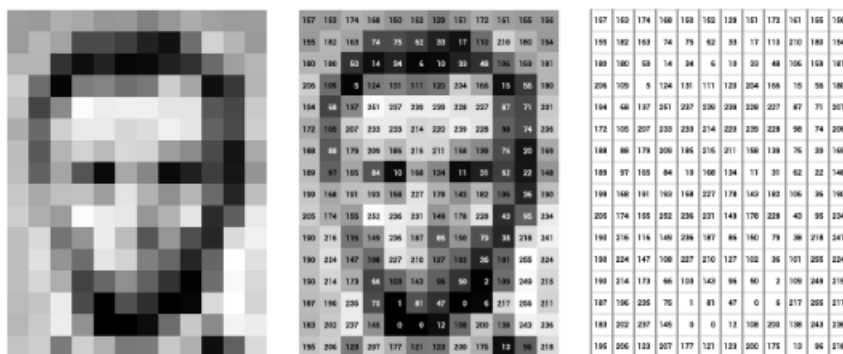
۱. مزایای استفاده از CNN در تسک‌های پردازش تصویر چیست؟
۲. آنچه در فرآیند Batch Normalization رخ می‌دهد را به صورت مختصر توضیح دهید و مزایای استفاده از آن در شبکه‌های عصبی عمیق را نیز بیان نمایید.
۳. توضیح دهید چرا Residual Learning از محو شدگی گرادیان (Vanishing Gradient) جلوگیری می‌کند؟

بخش دوم؛ پیاده‌سازی:

در این تمرین ضمن آشنایی با شبکه‌های عصبی پیچشی^۱، به حل مسئله‌ی رنگ‌آمیزی تصاویر خاکستری (Image Colorization) می‌پردازیم.

❖ Image Colorization:

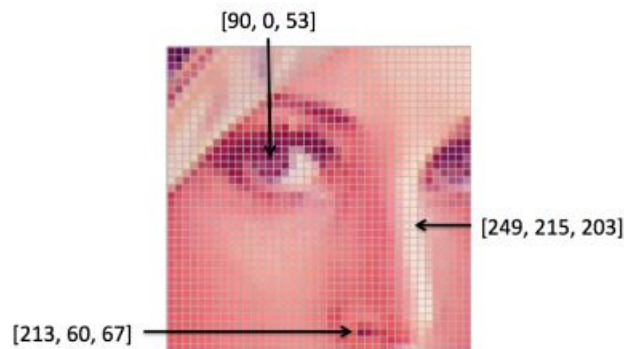
همانطور که می‌دانید، هر تصویر ماتریسی است که هر یک از درایه (پیکسل)های آن مقدار عددی‌ای در بازه‌ی [0,255] دارند.



تصویر ۱- تصویر grayscale (<https://ai.stanford.edu/~syueung/cvweb/tutorial1.html>)

در تصاویر خاکستری یا grayscale (همانطور که در تصویر ۱ نشان داده شده است) هر پیکسل تنها از یک مقدار عددی تشکیل شده است؛ به عبارتی ما با یک آرایه‌ی دوتایی روبه‌رو هستیم. این در حالی است که در هر تصویر رنگی (RGB)، هر پیکسل به صورت یک آرایه‌ی سه‌تایی می‌باشد که هر مقدار آن به یکی از کانال‌های قرمز، سبز و یا آبی تعلق دارد (تصویر ۲).

¹ Convolutional Neural Network (CNN)



تصویر ۲- تصویر RGB (<https://ai.stanford.edu/~syyeung/cvweb/tutorial1.html>)

* برای آشنایی بیشتر با عملیات کانولوشن و انواع فیلترها می‌توانید از این [لینک](#) استفاده نمایید.

در Image Colorization، هدف کلاس‌بندی هر پیکسل از تصویر grayscale با یکی از رنگ‌هایی است که (به عنوان لیبل یا کلاس) در پوشه‌ی پروژه، با نام car_colours_kmeans24.npy قرار داده شده است. بنابراین در انتها انتظار می‌رود که تصویر خروجی، یک تصویر رنگی باشد.

:Dataset

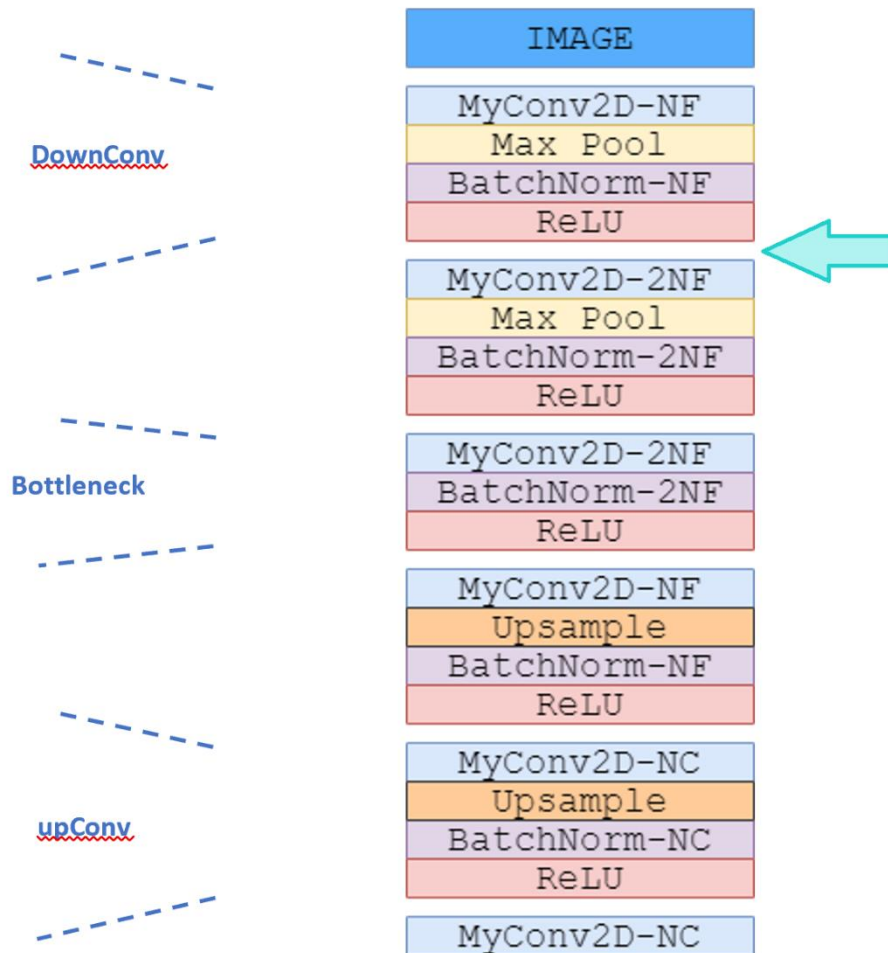
برای این تسک، مجموعه دادگان CIFAR-10 و کلاس automobile در نظر گرفته شده است. این دادگان شامل ۶۰۰۰۰ تصویر رنگی به ابعاد ۳۲×۳۲ در ۱۰ کلاس است (۵۰۰۰۰ تصویر آموزش و ۱۰۰۰۰ تصویر آزمون). برای به دست آوردن دادگان متناسب با تسک مطرح شده، نیاز به اجرا کردن بلوک مربوطه در نوت‌بوک CNN.ipynb می‌باشد. برای

استفاده از این مجموعه داده، توابع مربوط برای تبدیل تصاویر اصلی به تصاویر ورودی مد نظر در صورت پروژه و لیبل متناسب با آن در فایل `utils.py` قرار داده شده‌اند و نیازی به پیاده‌سازی نبوده و تنها کافی است تا فراخوانی شوند (منظور تبدیل تصاویر RGB به تصاویر Grayscale در تصاویر ورودی و تبدیل لیبل دسته‌بندی تصاویر (ماشین، هواپیما و ...) به لیبل متناظر با رنگ هر پیکسل برای پیکسل‌های هر تصویر ورودی می‌باشد).

:Network architecture

برای انجام تغییرات لازم بر روی تمامی مدل‌ها، کلاس‌های مربوطه در فایل `model.py` قرار داده شده‌اند. در این بخش لازم است مدلی مشابه معماری داده شده در تصویر ۳ در کلاس `BaseModel` پیاده‌سازی شود. □ قابل ذکر است با توجه به استفاده از تابع فعال‌سازی ReLU، پیش‌بینی کلاس مربوط به هر پیکسل به صورت دوتایی (binary) خواهد بود. بنابراین به تعداد کلاس‌های رنگ در لایه‌ی آخر فیلتر خواهیم داشت که در تصویر ۳ با `num_color(nc)` نشان داده شده است (خروجی برای هر پیکسل آرایه‌ای به سبب `nc` است). با توجه به نتایجی که حاصل می‌شود، بهترین مقدار را برای `num_filters(nf)` مشخص نموده و گزارش نمایید. □ با تعیین مقدار `nc` و `nf`، تعداد کانال‌های خروجی و ورودی برای تمامی لایه‌ها مشخص می‌شود. قابل ذکر است که نیاز است تا اندازه‌ی کرنل لایه کانولوشن را نیز تعیین نمایید □ (به عنوان مثال عدد ۳).

استفاده از `early stopping` برای آموزش امتیازی می‌باشد. □



تصویر ۳- معماری شبکه‌ی Base Model

فرضیات:

- اندازه کرنل مربوط به max pooling را 2×2 در نظر بگیرید.
- scale-factor مربوط به لایه‌ی upsample را ۲ در نظر بگیرید.
- برای تمامی مدل‌ها از تابع هزینه‌ی cross-entropy استفاده نمایید.

- تابع plot برای نمایش تصاویر در فایل utils.py، تصویر رنگی، تصویر خاکستری و خروجی شبکه را به عنوان ورودی می‌گیرد و می‌توان از آن برای نمایش یا ذخیره‌ی دادگان آزمون یا آموزش در هر ایپاک استفاده نمود. این تابع به صورت زیر فراخوانی می‌شود.

```
plot(xs, ys, predicted.cpu().numpy(), colours,
     save_dir + '/train_%d.png' % epoch,
     args.visualize)
```

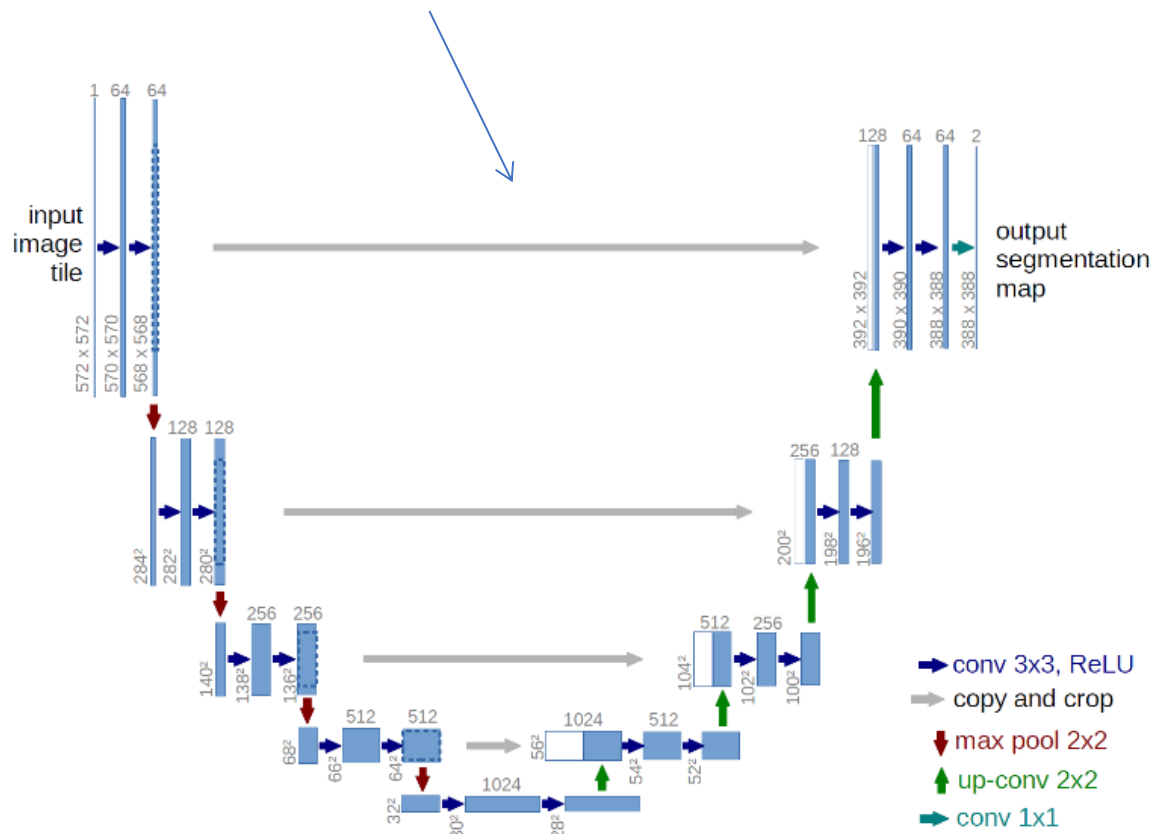
ابعاد Feature Map مربوط به لایه‌ی مشخص شده (با فلش) در شکل ۳ را به دست آورید و روش محاسبه را نیز بیان نمایید □.

بعد از آموزش شبکه، نتایج خود را با رسم نمودار تحلیل نمایید □.

❖ Skip connection:

در کلاس با Residual Block آشنا شدید. در این تمرین می‌خواهیم از connection ای متفاوت که در معرفی شبکه U-Net [1] آمده است استفاده نماییم. در این connection، ارتباط بین لایه n ام و لایه‌ی اول، لایه‌ی $(n-1)$ ام و لایه‌ی دوم، و... خواهد بود. به عبارتی این لایه‌ها با هم concatenate می‌شوند. نحوه‌ی عملکرد این connection در تصویر ۴ قابل مشاهده می‌باشد. این نوع ارتباط را بر مدل بخش قبل اعمال کنید □ (کلاس CustomUNet در فایل main.py).

نتایج دو بخش را با هم مقایسه و تحلیل نمایید □.

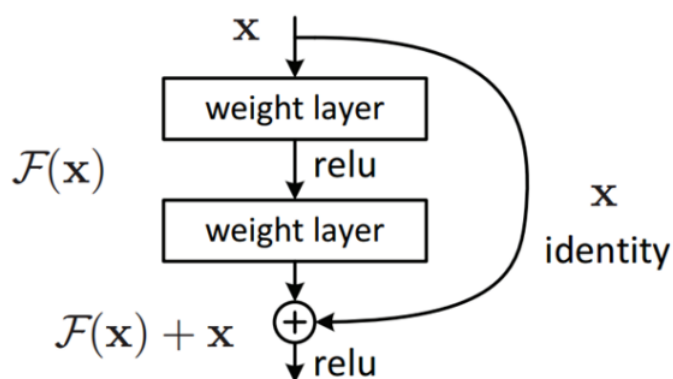


تصویر ۴- شبکه‌ی U-Net

❖ بخش امتیازی:

با توجه به آشنا بودن با ساختار Residual Block، آن را بر شبکه‌ی CustomUNet اعمال کنید □ (تبدیل DownConv، UpConv، Bottleneck به residual block. با توجه به اینکه در residual block دو لایه کانولوشنی وجود دارد، احتمالا نیاز به اضافه کردن لایه conv می‌باشد).

نتایج به دست آمده را تحلیل نموده و نمودار مربوط به آن را نیز رسم نمایید □.



تصویر ۵. Residual block

موارد تحویل:

علاوه بر مواردی که به منظور راحتی شما با Check Box در متن پروژه مشخص شده‌اند، لازم است موارد زیر را نیز در هنگام نوشتن گزارش و تحویل پروژه مد نظر داشته باشید.

- فایل‌هایی که در اختیارتان قرار گرفته است صرفاً الگویی است که مسیر پیاده‌سازی را نشان می‌دهد و قابل تغییر می‌باشند. اما لازم است مواردی که در این فایل‌ها مشخص شده‌اند حتماً پیاده‌سازی شوند.

- فایل‌های `model.py`، `train.py` و `torch_helper.py` در کنار نوت‌بوک `CNN.ipynb` نیاز به تکمیل دارند.

- می‌توانید فایل‌های اولیه این تمرین به همراه توضیحات تکمیلی را از ریپازیتوری درس دریافت نمایید. پس از تکمیل کدها، لازم است گزارشی شامل خروجی‌ها به همراه تصاویر نمودارهای رسم شده تهیه نمایید. فایلی که آپلود می‌کنید باید شامل تمامی کدها و گزارش شما باشد و در پوشه `docs/` است قرار گیرد.

- **نکته مهم:** علاوه بر قابل اجرا بودن کد شما، نیاز است تا خروجی‌هایتان نیز در نوت‌بوک موجود باشد. عدم ذخیره‌ی خروجی هر قسمت، سبب از دست دادن نمره آن قسمت خواهد شد.

🚩 نحوه ارزیابی:

تکمیل توابع و اسکریپت‌های پایتون و آموزش مدل‌های خواسته شده ← ۴۰ نمره
بررسی تنظیمات مختلف برای نتیجه‌ی بهتر (num features, تعداد epochها, learning rate, optimizers و ...) ← ۱۵ نمره
پاسخ به سوالات مطرح شده در بخش اول (۳ سوال) ← ۱۵ نمره
گزارشی از نتایج و نمودارها در نوت‌بوک به همراه فایل PDF تحلیل نتایج ← ۳۰ نمره
استفاده از early stopping برای آموزش (امتیازی) ← ۵ نمره
بخش امتیازی ← ۱۵ نمره

در صورت وجود هرگونه پرسش و ابهام می‌توانید سوالات خود را در گروه تلگرامی این درس و یا با ارسال ایمیل به یکی از ایمیل‌های زیر مطرح نمایید.

gheshlaghitara@gmail.com

mohammad99hashemi@gmail.com

RamezaniKh.Motahareh@gmail.com

😊 موفق باشید

مراجع: 

- [1] Olaf Ronneberger, Philipp Fischer, Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015.