

به نام خدا



دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده برق و کامپیوتر



## شبکه‌های عصبی مصنوعی و یادگیری عمیق

مینی پروژه شماره ۳

دی ۹۹

## فهرست سوالات

- 3..... توضیحات کلی مینی پروژه
- 5..... Variational Autoencoder – ۱ سوال
- 7..... Conditional GANs – ۲ سوال
- 9..... Semantic Image Synthesis with SPADE – ۳ سوال
- 11..... CycleGAN – ۴ سوال
- 12..... ۵ – آشنایی با مقالات مربوط

## توضیحات کلی مینی پروژه

هدف این تکلیف، آشنایی شما با شبکه‌های VAE و GAN است. در سوال اول از شما خواسته می‌شود که یک شبکه VAE را پیاده‌سازی کنید. در سوالات دوم، سوم و چهارم هم با تعدادی از شبکه‌های معروف GAN آشنا می‌شوید؛ و از خواسته شده که آن‌ها را هم پیاده‌سازی کنید. سوال آخر هم سوالی از نوع Paper Review است و می‌توانید به انتخاب خودتان یکی از مقالات را مطالعه کرده و به سوالات مربوط پاسخ دهید.

با توجه به اینکه آموزش دادن شبکه‌های GAN به سخت‌افزار قدرتمندی احتیاج دارد، پیشنهاد ما آن است که در محیط [Google Colab](https://colab.research.google.com/) کد خود را پیاده‌سازی کنید و از GPU هایی که این محیط در اختیارتان می‌گذارد، استفاده کنید.

در اجرای آموزش مدل‌های مسائل، تعداد ایپاک‌های آموزش را خودتان و بر اساس مطلوب بودن خروجی، انتخاب کنید. دقت کنید که هدف رسیدن به دقتی برابر با دقت مقالات نیست و چیزی که اهمیت دارد پیاده‌سازی صحیح شبکه‌ها است.

در همه مسائل از شما توضیح ساز و کار معماری و همچنین نتایج پیاده‌سازی آن خواسته شده که در ادامه نحوه ارائه توضیحات برای شما شرح داده می‌شود.

### توضیح ساز و کار معماری :

برای هر معماری که از شما پیاده‌سازی آن خواسته می‌شود، لازم است مقاله مربوط به آن را مطالعه کنید و در حدود چند پاراگراف در مورد طراحی و سازوکار معماری آن توضیح دهید.

### نتایج مورد نیاز بعد اجرای هر مدل :

(۱) نمونه تصاویر تولید شده :

برای هر مدلی که در سوالات آموزش می‌دهید، نتایج را طی ۵ ایپاک مختلف از میان همه ایپاک‌های آموزش، نمایش دهید. اینکه در کدام ایپاک نتایج تان را نمایش می‌دهید بر عهده خودتان است، ولی نمایش محسوس روند پیشرفت شبکه و بهبود کیفیت تصاویر تولیدی، کاملاً ضروری است.

تصاویر تولیدی باید به صورت ۱۰ در ۱۰ مانند تصویر یک باشند.



تصویر ۱. نمونه تصاویر برای داده‌های MNIST

۲) نمودار Loss بر حسب ایپاک :

بسته به مدل شبکه انتخابی، نمودار دقت یا Loss را به ازای هر ایپاک آموزش رسم کنید.

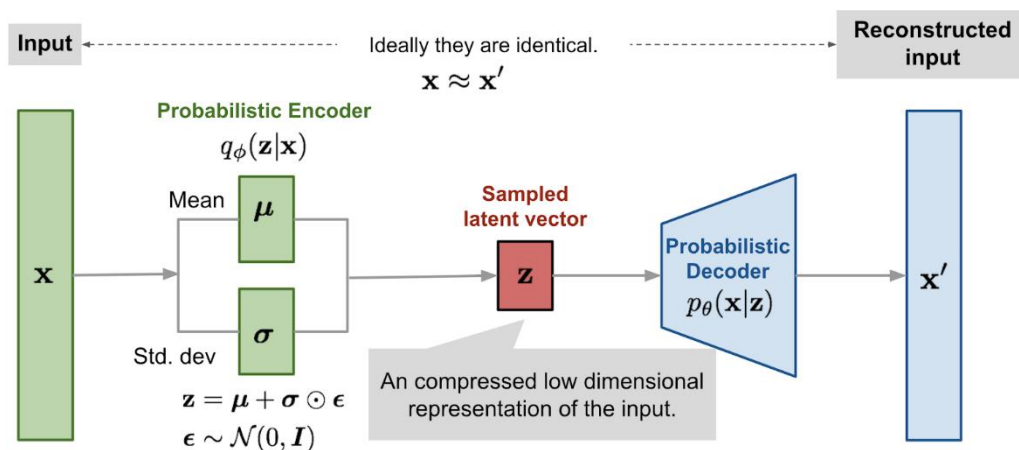
### توضیح کد:

همچنین در این سوالات از شما خواسته می‌شود که روند پیاده‌سازی در فایل Jupyter Notebook کد خودتان را توضیح دهید. توجه کنید که نیازی به توضیح همه جزئیات نیست و توضیح روند کلی پیاده‌سازی مدل کفایت می‌کند.

با توجه به این توضیحات، دیگر نیازی به ذکر روند پیاده‌سازی مدل در گزارش کار نیست.

## سوال ۱ – Variational Autoencoder

شبکه های Variational Autoencoder که به اختصار VAE نامیده میشوند، از جمله مدل های قدرتمند برای تولید داده هستند. همانطور که از اسم این شبکه ها برمی آید، معماری آن ها مانند Autoencoder های عادی دارای دو قسمت/شبکه encoder و decoder است. در یک Autoencoder معمولی هر قسمت یک نداشت را تقریب میزند. قسمت encoder یک نداشت از بعد بالاتر (مثلا تصویر) به بعدی کمتر (latent space) را تقریب میزند در حالی که decoder تلاش می کند نداشت وارون را برای بازیابی داده اولیه پیدا کند.



تصویر 2. معماری VAE

استفاده از Autoencoder های معمولی برای تولید داده مشکل ساز است، زیرا که داده ها در فضای ثانویه به شکل دلخواه قرار گرفته شده اند. این آزادی بیش از حد از دو جهت مشکل ساز است. اول آن که با تغییرات اندک در فضای ثانویه، داده تولید شده میتواند بسیار متفاوت باشد بنابراین شبکه نسبت به نویز مقاوم نیست. دوم آن که به دلیل دلخواه بودن فضای خروجی هیچ کنترلی بر داده تولید شده وجود ندارد. برای حل اولین مشکل، در شبکه های VAE به جای آنکه نگاشته ها بین دو فضا با ابعاد متفاوت ساخته شود، بین فضای اولیه و یک توزیع احتمال در فضای ثانویه ساخته میشود. با این روش نه تنها داده های مشابه در فضای ثانویه به یکدیگر نزدیک میشوند، بلکه به دلیل نمونه برداری که در فضای ثانویه انجام میشود، داده های تولید شده در برابر تغییرات کوچک حساس نیستند. برای حل مشکل دوم، از شبکه های Conditional VAE استفاده میشود. در این شبکه ها، با استفاده از شرطی کردن توزیع احتمال ورودی، فضای ثانویه و خروجی بر پارامترهای اضافی می توان داده های تولید شده را کنترل کرد.

برای مطالعه بیشتر در مورد این شبکه ها می توانید به [اینجا](#) و [اینجا](#) مراجعه کنید. همچنین مقاله مربوط به VAE را می توانید از [این منبع](#) دریافت کنید. مقاله CVAE نیز در [این منبع](#) در دسترس است.

**الف)** شبکه VAE را بر روی مجموعه داده MNIST پیاده سازی کنید. برای ساختارهای decoder و encoder نیازی نیست از معماری خاصی پیروی کنید، تنها شرطی که در معماری شبکه باید رعایت کنید آن است که بعد فضای ثانویه ۲ باشد.

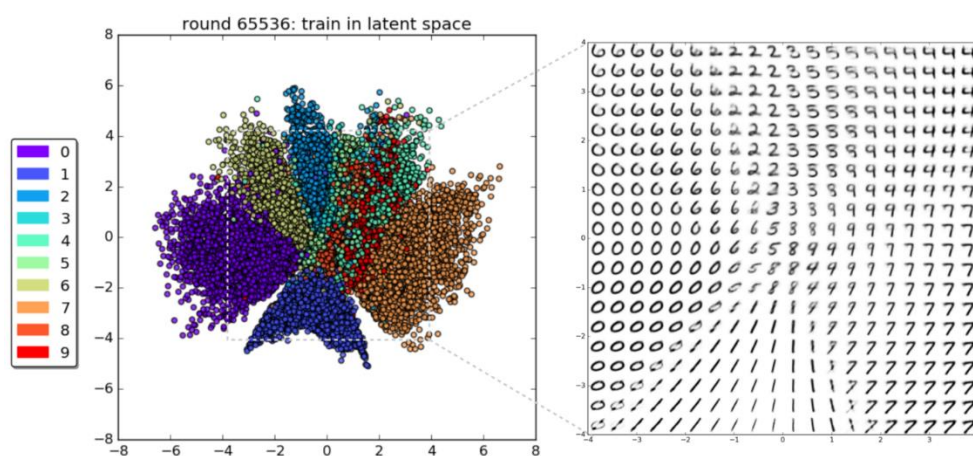
**ب)** از چه تابع هزینه ای استفاده کرده اید؟ لزوم استفاده از تابع KL divergence را توضیح دهید.

**ج)** مجموعه داده های MNIST را با استفاده از شبکه VAE ای که آموزش داده اید به فضای ثانویه انتقال دهید و Scatter Plot آن ها را رسم کنید. هدف در این قسمت بررسی چگونگی قرار گرفتن توزیع داده های هر کلاس است، بنابراین داده های هر کلاس باید با رنگ و نام از روی شکل قابل تمیز باشند.

**د)** پس از آن که داده ها را به فضای ثانویه انتقال دادید، می توانید حدود پراکندگی این داده ها را به سادگی بدست آورید ( یعنی بیشترین مقدار و کمترین مقداری که داده ها در هر بعد اختیار کرده اند در دسترس است). از decoder آموزش داده شده استفاده کنید و یک grid با اندازه دلخواه ( مثلا  $10 \times 10$ ) از داده های بازیابی شده در این حدود رسم کنید (مانند آنچه در تصویر 3 می بینید).

**ه)** شبکه CVAE را با استفاده از مجموعه داده MNIST آموزش دهید. همانند قسمت ( ج ) داده ها را به فضای ثانویه انتقال دهید و Scatter Plot آن ها را رسم کنید. تفاوت میان نتیجه این قسمت و قسمت ج را توضیح دهید.

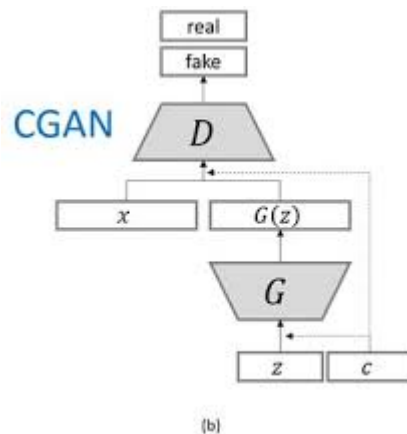
**و)** به دلخواه خود ۲ برچسب (شماره کلاس) را انتخاب کنید، و با شرطی کردن decoder بر هر یک از آنها همانند قسمت ( د ) داده های تولید شده را در یک grid نشان دهید.



تصویر 3. خروجی نمونه برای شبکه VAE

## سوال ۲ - Conditional GANs

در این بخش به بررسی Conditional GAN ها و Deep Convolutional GAN ها می پردازیم. شبکه‌های DCGAN از شبکه‌های عصبی بر پایه GAN هستند که در طراحی معماری آن‌ها از لایه‌های Convolutional استفاده میشود. مقاله مربوط به شبکه‌های DCGAN را میتوانید از [اینجا](#) مشاهده نمایید. شبکه بعدی که در این سوال مورد بررسی قرار می گیرد، شبکه‌های CGAN هستند، در این شبکه‌ها، هدف استفاده از اطلاعاتی اضافه تر از تصاویر برای تولید خروجی‌های با کیفیت تر استفاده می‌شود. در مقاله CGAN که در سال ۲۰۱۴ منتشر شد، علاوه بر تصاویر، Label مربوط به آن‌ها نیز به شبکه داده می‌شد تا بتوان روی خروجی تولید شده، کنترل داشت. این مقاله را می‌توانید از [اینجا](#) مشاهده کنید. معماری CGAN را در تصویر پایین مشاهده می‌کنید.



تصویر 4. معماری کلی CGAN

**الف)** سازوکار شبکه CGAN را شرح دهید.

**ب)** در این بخش با استفاده از دیتاست Fashion MNIST ترکیب این دو شبکه را پیاده‌سازی کنید و نتایج را در گزارش تان بیاورید.



تصویر 5. نمونه خروجی برای هر کلاس

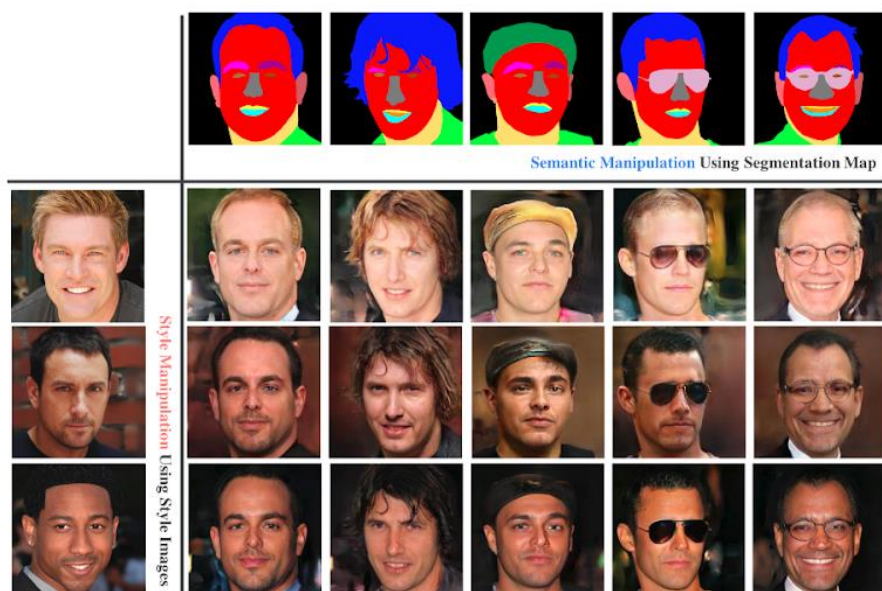
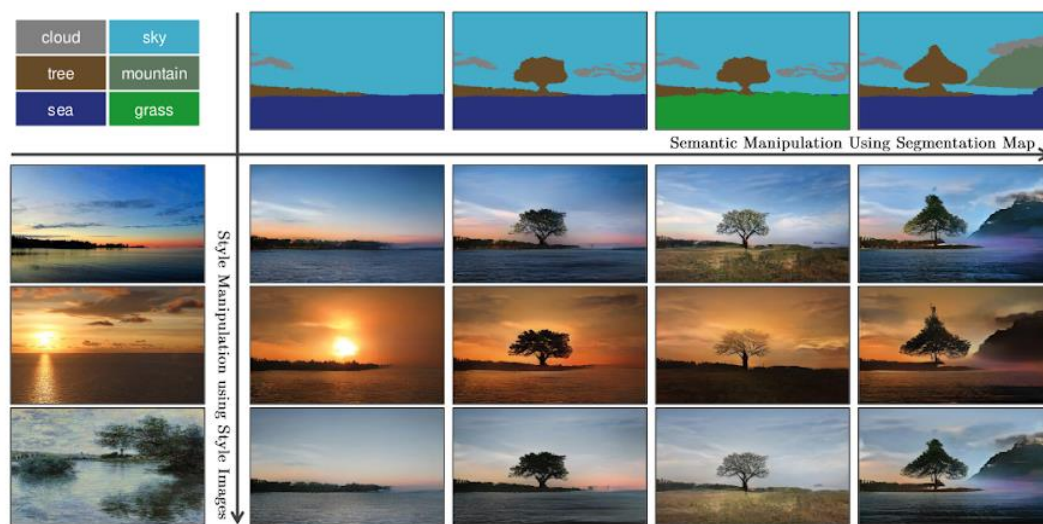
ج) در این بخش می خواهیم تاثیر Optimizer را روی عملکرد شبکه بررسی کنیم. برای این کار دو بار شبکه را با Adam و RMSprop آموزش دهید و خروجی را به ازای تعداد ایپاک برابر مقایسه کنید. راجع به تفاوت عملکرد این دو تحقیق کنید و نتایج را بررسی و گزارش کنید.

د) در این بخش می خواهیم تاثیر Loss function را روی عملکرد شبکه بررسی کنیم. ابتدا در مورد توابع مختلفی که برای آموزش GAN ها مورد استفاده قرار می گیرند تحقیق کرده و دو مورد را انتخاب کنید. شبکه را با هر دو تابع آموزش دهید و نتایج را به ازای تعداد ایپاک برابر مقایسه کنید. علاوه بر این، نمودار loss را برای هر دو رسم کنید و نتایج حاصل را بررسی کنید.



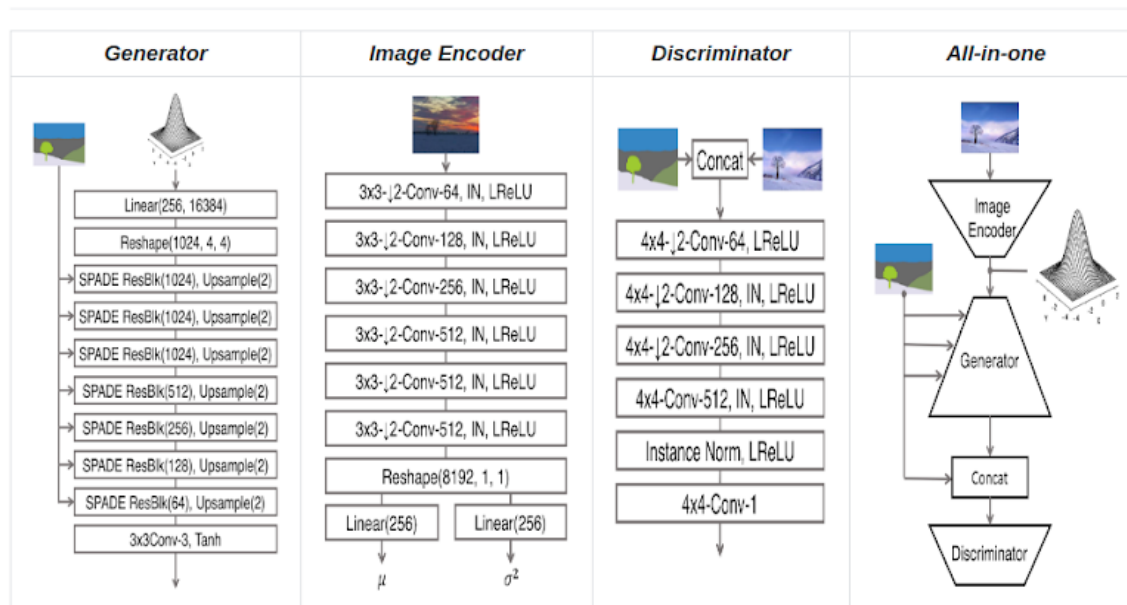
### سوال ۳ - Semantic Image Synthesis with SPADE

در این بخش به بررسی و پیاده‌سازی مقاله Semantic Image Synthesis with SPADE می‌پردازیم. در این [مقاله](#) تصاویر واقعی از روی ورودی‌هایی به شکل مشخص تولید می‌شود. در ادامه می‌توانید نمونه‌هایی را مشاهده کنید.



تصویر 6. نمونه‌های خروجی تولیدی مدل پیشنهادی روی دو دیتاست مختلف

ساختار کلی این شبکه به شکل زیر است که برای درک جزئیات بیشتر به مقاله‌ای که لینک آن در اختیارتان قرار گرفته است مراجعه کنید.



تصویر 7. ساختار کلی شبکه

**الف)** طرز عملکرد کلی شبکه و همچنین ساختار generator ایی که در آن از مدل پیشنهادی SPADE استفاده شده است را شرح دهید.

**ب)** شبکه را با استفاده از یکی از دیتاست‌هایی که مناسب اند مانند CelebA-HQ یا COCO-Stuff پیاده سازی کنید و نتایج را مانند تصویر 6 در گزارش خود نمایش دهید. همچنین مراحل کلی پیاده سازی را در فایل Notebook خود توضیح دهید. توصیه می‌شود کلیه روند پروژه را در در Google Colab انجام دهید. همچنین استفاده از وزن‌های از پیش آموزش داده شده در صورت وجود بلامانع است.

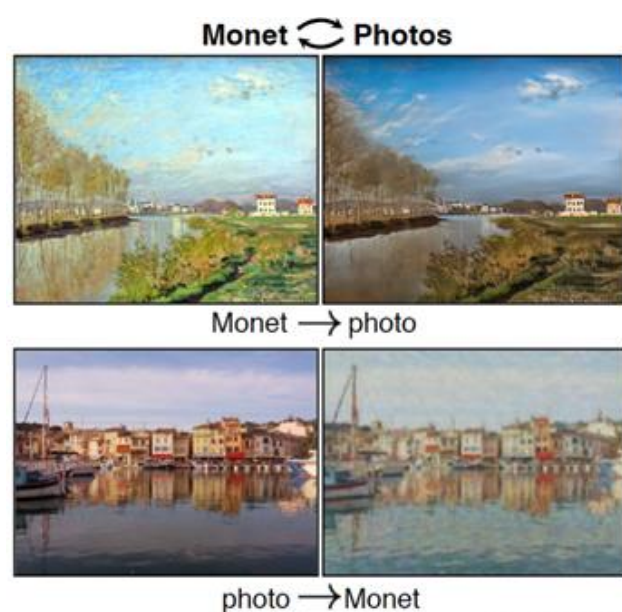
لینک دیتاست CelebA-HQ :

<https://github.com/switchablenorms/CelebAMask-HQ>

نکته: برای دانلود دیتاست‌ها، در صورت دردسترس بودن روی Google Drive، می‌توانید مستقیماً و بدون نیاز به دانلود، به درایو گوگل خود اضافه و استفاده کنید.

## سوال ۴ - CycleGAN

یکی از کاربردهای شبکه‌های GAN، تبدیل تصویر به تصویر از یک مجموعه به مجموعه دیگر و برعکس است. شبکه CycleGAN، یک توسعه از معماری شبکه‌های GAN است که بدین منظور استفاده می‌شود. این شبکه سعی دارد بصورت همزمان دو مدل Generator و دو مدل Discriminator را آموزش دهد. یکی از Generator ها، تصاویر را از مجموعه اول دریافت می‌کند و سعی می‌کند تصاویر مجموعه دوم را بسازد. Generator دوم، عکس این عمل را انجام می‌دهد. سپس به کمک Discriminator ها تعیین می‌شود که هر کدام از این Generator ها، تا چه حد در کار خود موفق بوده‌اند. بعد از ارزیابی، مدل‌های Generator به روز می‌شوند. مقاله مدل CycleGAN را می‌توانید از [اینجا](#) دانلود کنید.



تصویر 8. نمونه تصاویر تولید شده برای تبدیل نقاشی‌های Claude Monet به تصاویر واقعی و بالعکس

الف) ساز و کار طراحی معماری CycleGAN را بصورت خلاصه شرح دهید.

ب) شبکه CycleGAN را با استفاده از دیتاست monet2photo پیاده‌سازی کنید و نتایج را همانند تصویر بالا در گزارش خود نمایش دهید.

ج) مراحل پیاده‌سازی را در فایل Notebook خود توضیح دهید. سعی کنید کد خود را به صورت ماژولار بنویسید تا کار توضیح دادن قسمت‌های مختلف شفاف‌تر باشد.

## سوال 5 – آشنایی با مقالات مربوط

در این سوال قصد داریم تا با کاربردها، ساختارها و چالش‌های شبکه‌های عصبی مولد رقابتی در مقالات جدید آشنا شویم. به این منظور، هر گروه باید یکی از چهار مقاله پیشنهادی در این سوال را انتخاب کرده و مواردی که در ادامه شرح داده شده است را برای آن مقاله خاص تهیه کند.

برای این سوال باید یک خلاصه از مقاله تهیه نمایید. در این خلاصه باید به موارد کلیدی مانند معماری شبکه، تابع هزینه و همچنین مجموعه داده استفاده شده برای آموزش شبکه، اشاره کنید. همچنین پیشینه مقاله، دست‌آورد نویسندگان و مسیر پیشنهادی برای ادامه تحقیقات (در صورت وجود در متن مقاله) در آن موضوع را بیان کنید. در خلاصه خود، هر قسمت را به صورت مجزا در یک یا چند پاراگراف بنویسید تا از یک‌دیگر قابل تمیز باشند.

پس از مطالعه و تحلیل مقاله باید با استفاده از یک شبکه آموزش دیده شده، نمونه‌ای از عملکرد مدل را نمایش دهید. برای این قسمت سعی کنید اگر پیاده‌سازی اصلی مقاله استفاده کنید تا خروجی‌های شما بیشترین شباهت را به نتایج مقاله داشته باشد. برای هر مقاله، مواردی که باید گزارش شود به صورت مجزا اشاره شده است.

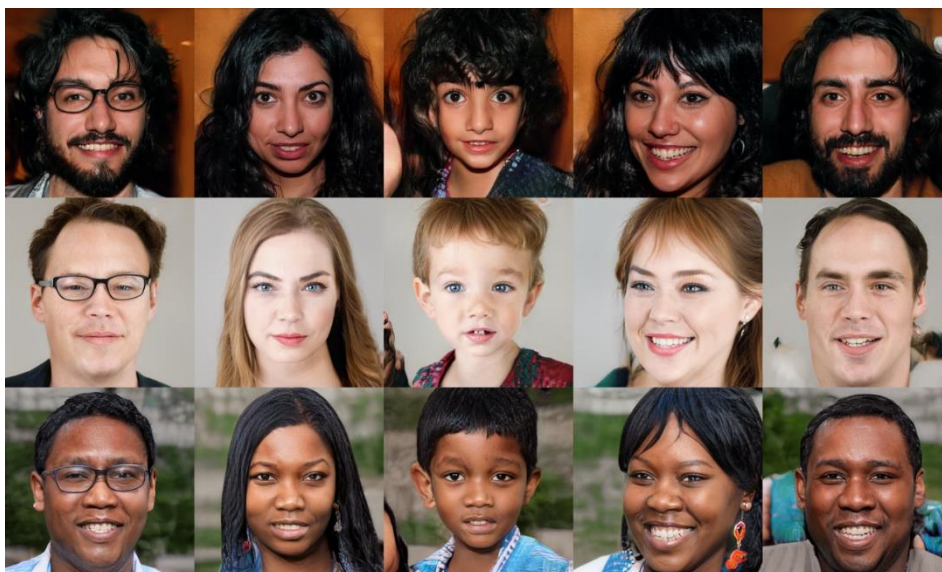
### InterFaceGAN -1

در [این](#) مقاله بر روی تغییر چهار ویژگی از تصاویر چهره اشخاص کار شده است. این ویژگی‌ها شامل جنسیت، سن، ژست و افزودن عینک است. برای نمایش نتایج باید دو تصویر با جنسیت‌های مخالف را انتخاب کنید و تصویری شبیه به تصویر ۱ در مقاله را ایجاد کنید.

### StyleGAN -2

ساخت تصویر جدید از چهره انسان از جمله کاربردهای شبکه GAN است، به طور مثال در تصویر زیر هیچکدام تصویر واقعی یک انسان نیست و با استفاده از شبکه GAN و یک مجموعه داده تولیدشده است، معماری PGGAN و StyleGAN از جمله معماری‌هایی هستند که در تولید تصاویری با چهره انسان به کار می‌رود. مقاله StyleGAN در سال ۲۰۱۸ توسط محققان شرکت Nvidia منتشر شد. از [اینجا](#) می‌توانید مقاله مورد نظر را دانلود کنید.





تصویر 9. نمونه تصاویر تولید شده توسط StyleGAN

### StackGAN -3

از کاربردهای خیلی معروف و جذاب GAN می توان به تبدیل متن به تصویر اشاره کرد، در این حالت یک متن به شبکه داده می شود و متناسب با متن، شبکه GAN به شما یک تصویر را به عنوان خروجی می دهد. به طور مثال در تصویر زیر یک متن شامل توصیف یک پرنده را دریافت می کند و آن را به تصویر می کشد، یکی از معروف ترین مدل ها در این حوزه مدل StackGAN است که مقاله آن را می توانید از [اینجا](#) دانلود نمایید .

Text description	This bird is red and brown in color, with a stubby beak	The bird is short and stubby with yellow on its body	A bird with a medium orange bill white body gray wings and webbed feet	This small black bird has a short, slightly curved bill and long legs	A small bird with varying shades of brown with white under the eyes	A small yellow bird with a black crown and a short black pointed beak	This small bird has a white breast, light grey head, and black wings and tail
64x64 GAN-INT-CLS							
128x128 GAWWN							
256x256 StackGAN							

تصویر 10. نمونه تصاویر تولید شده توسط StackGAN

#### PULSE: Self-Supervised Photo Up sampling via Latent Space Exploration –4 of Generative Models

این [مقاله](#) برای یک تصویر با کیفیت پایین، تصویر مناسب با کیفیت بالا که توسط یک مدل مانند StyleGAN تولید می‌شود را پیدا کند.



تصویر 11. نمونه عکس‌های تولید شده از تصاویر ورودی با کیفیت پایین.

## نکات:

- مهلت تحویل این مینی پروژه ۱۲ بهمن است.
- گزارش را در قالب تهیه شده که روی صفحه درس در Elearn بارگذاری شده، بنویسید.
- گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است. لطفاً تمامی نکات و فرض‌هایی که برای پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید را در گزارش ذکر کنید.
- در گزارش خود برای تصاویر زیرنویس و برای جداول هم بالانویس اضافه کنید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست. اما باید نتایج بدست آمده را گزارش و تحلیل کنید.
- دستیاران آموزشی ملزم به اجرا کردن کدهای شما نیستند. بنابراین هرگونه نتیجه و یا تحلیلی که در شرح سوال از شما خواسته شده است را به طور واضح و کامل در گزارش بیاورید. در صورت عدم رعایت این مورد، بدیهی است که از نمره تمرین کسر می‌شود.
- در صورت مشاهده تقلب امتیاز تمامی افراد شرکت‌کننده در آن، 100- لحاظ می‌شود.
- برای انجام تمرین‌ها و مینی پروژه‌ها، تنها زبان برنامه نویسی مجاز Python است.
- استفاده از کدهای آماده برای تمرین‌ها به هیچ وجه مجاز نیست. اما برای مینی پروژه‌ها فقط برای قسمت‌هایی از کد و به عنوان راهنمایی برای پیاده‌سازی، می‌توانید از کدهای آماده استفاده کنید.
- برای این تکلیف، تنها سه روز مهلت ارسال با تاخیر در نظر گرفته شده است. پس از مهلت اصلی، تا تاریخ ۱۵ بهمن، با جریمه ۳۰ درصدی کسر امتیاز، می‌توانید کدها و گزارش خود را ارسال کنید.
- لطفاً گزارش، فایل کدها و سایر ضمایم مورد نیاز را با فرمت زیر در سامانه مدیریت دروس بارگذاری نمایید.

PROJECT3\_[Lastname]\_[StudentNumber].zip

- در صورت وجود هرگونه ابهام یا مشکل می‌توانید از طریق رایانامه‌های زیر با دستیاران آموزشی مربوطه، آقایان سجاد پاکدامن ( سوال ۱)، سینا شریفی ( سوال ۲)، امیرحسین قاسم آبادی (سوال ۳) و محسن معماریان (سوال ۴) در تماس باشید:

- [sj.pakdaman@ut.ac.ir](mailto:sj.pakdaman@ut.ac.ir)
- [sharifi.sina1377@gmail.com](mailto:sharifi.sina1377@gmail.com)
- [amirho3eigh98@gmail.com](mailto:amirho3eigh98@gmail.com)
- [meamarian.mohsen@ut.ac.ir](mailto:meamarian.mohsen@ut.ac.ir)