

## ԲՈՎԱՆԴԱԿՈՒԹՅՈՒՆ

ԲՈՎԱՆԴԱԿՈՒԹՅՈՒՆ .....	1
ՆԵՐԱԾՈՒԹՅՈՒՆ .....	2
ԽՆԴՐԻ ԴՐՎԱԾՔԸ .....	3
ԳԼՈՒԽ 1. ՏԵՍԱԿԱՆ ԱՌԸՆՉՈՒԹՅՈՒՆՆԵՐ.....	4
ԳԼՈՒԽ 2. ԾՐԱԳՐԻ ԻՐԱԿԱՆԱՑՈՒՄ .....	9
2.1 ԱԼԳՈՐԻԹՄԻ ՀԻՄՆԱԿԱՆ ՔԱՅԼԵՐԸ.....	9
2.2 ԾՐԱԳՐԻ ԻՐԱԿԱՆԱՑՄԱՆ ԲԼՈԿ ՍԽԵՄԱ.....	11
ԱՐԴՅՈՒՆՔՆԵՐ .....	17
ԵԶՐԱԿԱՑՈՒԹՅՈՒՆ.....	20
ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ.....	21

## ՆԵՐԱԾՈՒԹՅՈՒՆ

Թվային դարաշրջանի առաջընթացի հետ միասին, արհեստական բանականության (ԱԲ) կիրառությունը մտել է տարբեր ոլորտներ, այդ թվում՝ գեղարվեստական ստեղծագործությունների ստեղծման գործընթաց: Նորարարական մեթոդներից մեկը, որը փոխել է մեր պատկերացումները արվեստի ստեղծման ավտոմատացված հնարավորությունների մասին, հանդիսանում է Neural Style Transfer-ը (NST)՝ ներդրողական ոճի փոխանցման տեխնոլոգիան: Neural Style Transfer-ը նորարարական ալգորիթմ է, որը թույլ է տալիս արվեստագետներին և մշակողներին սինթեզել նոր պատկերներ՝ միավորելով երկու տարբեր պատկերների հատկանիշները: Այդ գործընթացը ներառում է պատկերի "բովանդակության" հիմքում ընկած հիմնական կառուցվածքների պահպանումը և դրա վրա մեկ այլ պատկերի "ոճի" կիրառումը: Այս տեխնոլոգիան լայն ճանաչում ստացավ հատկապես այն բանից հետո, երբ առաջին անգամ ներկայացվեց Google DeepDream նախագծի և Visual Geometry Group (VGG) ներդրողական ցանցի հիման վրա: NST-ի հիմնական նպատակը՝ բովանդակության և ոճի առանձնացված ներկայացման հնարավորությունն է, որտեղ բովանդակությունը ներկայացնում է պատկերի կառուցվածքը, իսկ ոճը՝ նրա գունային ու տեքստուրային առանձնահատկությունները:

## ԽՆԴՐԻ ԴՐՎԱԾՔԸ

Neural Style Transfer-ը (NST) մեթոդը հիմնված է կոնվոլյուցիոն նեյրոնային ցանցերի (Convolutional Neural Networks - CNNs) վրա, որոնք ի սկզբանե մշակվել են պատկերի ճանաչման և դասակարգման խնդիրների համար: NST-ի հիմնական խնդիրն է պարզել, թե ինչպես կարող ենք առանձին ներկայացնել պատկերի բովանդակությունը և ոճը: Խնդրի էությունը կայանում է նրանում, որ NST-ն պետք է կարողանա համատեղել պատկերների այս երկու բաղադրիչները այնպես, որ ստեղծված պատկերն ունենա սկզբնական բովանդակության կառուցվածքը և ընտրված ոճի գեղարվեստական արտահայտությունը:

## ԳԼՈՒԽ 1. ՏԵՍԱԿԱՆ ԱՌԸՆՉՈՒԹՅՈՒՆՆԵՐ

**Transfer Learning**-ը մեքենայական ուսուցման մեթոդ է, որտեղ արդեն մի dataset-ի վրա սովորած մոդելը օգտագործվում է այլ dataset-ի կամ խնդրի համար: Սա հատկապես օգտակար է, երբ տվյալների քանակը սահմանափակ է կամ երբ նոր խնդիրը ունի որոշակի նմանություն այն խնդրին, որի վրա մոդելը նախապես պատրաստվել է: Օգտագործվում է պատկերների մշակման մեջ՝ ImageNet-ի վրա սովորած մոդելը կարող է օգտագործվել փոքր տվյալների վրա օբյեկտների ճանաչման համար:

Transfer Learning-ը աշխատում է հետևյալ կերպ՝

- 1) Նախապես սովորեցված մոդել (Pre-trained Model): Օրինակ՝ ImageNet տվյալների վրա սովորեցված մոդելը, որը ճանաչում է տարբեր օբյեկտներ):
- 2) Fine-Tuning: Մոդելը հարմարեցվում է նոր խնդրի համար՝ փոքր փոփոխություններով: Սա կարող է լինել՝
  - Միայն վերջնական շերտերի փոփոխությամբ:
  - Ամբողջ ցանցի նորից ուսուցմամբ:

Հիմնական առավելություններ՝

- 1) Ժամանակի խնայողություն - Մոդելը չի սկսում զրոյից, ինչը կրճատում է ուսուցման ժամանակը:
- 2) Տվյալների պահանջի նվազում - Պահանջվում է ավելի քիչ տվյալ նոր խնդիրը լուծելու համար:
- 3) Արդյունավետություն - Հաճախ ապահովում է ավելի բարձր ճշգրտություն, հատկապես սահմանափակ տվյալների դեպքում:

**Gram** մատրիցը լայնորեն օգտագործվում է վեկտորների միջև ներքին կապերն ուսումնասիրելու համար: Այն ցույց է տալիս վեկտորների միջև սերտ հարաբերությունները: Gram մատրիցը ստացվում է վեկտորների արտադրյալների (dot product) հաշվարկից:

Gram մատրիցի օգտագործումը՝

1. Նեյրոնային ռճափոխության (Neural Style Transfer) մեջ: Gram մատրիցը օգտագործվում է պատկերների ռճային հատկությունները ներկայացնելու համար: Մատրիցը հաշվարկվում է feature map-ի համար՝ ստացված կոնվոլուցիոն շերտերից, և ներկայացնում է feature-ների համատեղ կախվածությունը: Օրինակ՝ ռճի կորուստը (style loss) չափելու համար կարելի է օգտագործել երկու պատկերների Gram մատրիցների տարբերությունը:
2. Kernel մեթոդներում (Support Vector Machines, PCA և այլն): Kernel մեթոդներում Gram մատրիցը ներկայացնում է տվյալների ներքին կառուցվածքը Kernel ֆունկցիայի կիրառման միջոցով:
3. Գծային անկախության ստուգում: Վեկտորների շարքը գծային անկախ է, եթե Gram մատրիցը ոչ-հատուկ (non-singular) է:

**Միջին քառակուսային սխալանքը** (Mean Squared Error, MSE) չափանիշ է, որն օգտագործվում է ստացված արժեքների և իրական արժեքների միջև տարբերությունը գնահատելու համար: Այն հաճախ օգտագործվում է մեքենայական ուսուցման մոդելների որակի գնահատման և օպտիմալացման համար, հատկապես՝ ռեգրեսիայի խնդիրներում: Եթե ունենք  $y_i$  իրական արժեքներ  $\hat{y}_i$  մոդելի ստացված արժեքներ,  $n$  տվյալների ընդհանուր թիվը, ապա MSE-ն հաշվարկվում է հետևյալ բանաձևով

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

որտեղ՝

- $y_i$  — իրական արժեքը,
- $\hat{y}_i$  — ստացված արժեքը,
- $n$  — տվյալների քանակ :

**TensorFlow Graph**-ը TensorFlow-ի հիմնական տարրերից մեկն է: Այն օգտագործվում է TensorFlow-ում՝ մաթեմատիկական գործողությունները, տվյալների հոսքերը ներկայացնելու և օպտիմալացնելու համար:

TensorFlow-ը գրաֆի տեսքով ներկայացնում է՝

1. Nodes: Ներկայացնում են գործողությունները (օրինակ՝ գումարում, բազմապատկում, մատրիցայի օպերացիաներ):
2. Edges: Ներկայացնում են տվյալների հոսքը նոդերի միջև: Սա կարող է լինել թվեր, տենսորներ կամ ֆայլերից կարդացված տվյալներ:

TensorFlow Graph-ը նախապես սահմանվում է և գործողությունները կատարվում են գրաֆը ներբեռնելուց հետո: Սա թույլ է տալիս օպտիմալացնել և հարմարեցնել գրաֆը հատուկ սարքավորումների համար (օրինակ՝ GPU կամ TPU):

TensorFlow-ն ապահովում է, որ յուրաքանչյուր նոդի մուտքերը և ելքերը ունենան հստակ տիպեր (օրինակ՝ թվային, տենսորային):

Կարող է գրաֆը օպտիմալացնել՝ չկիրառվող նոդերը հեռացնելով կամ գործողությունները վերակազմակերպելով՝ կատարողականը բարելավելու համար:

**SSIM**-ը (Structural Similarity Index Measure) պատկերների որակը գնահատելու մեթոդ է, որն օգտագործվում է երկու պատկերի միջև կառուցվածքային նմանությունը չափելու համար: Այն հաշվի է առնում ոչ միայն երկու պատկերի պիքսել առ պիքսել տարբերությունները, այլև այն, թե որքանով են նրանց կառուցվածքային հատկությունները նման: SSIM-ը հաճախ օգտագործվում է պատկերների վերականգնման, ռճափոխության և այլ համակարգերում՝ արդյունքների որակը գնահատելու նպատակով:

SSIM-ը հաշվի է առնում հետևյալ գործոնները.

1. Լուսավորություն (Luminance): Պատկերների ընդհանուր լուսավորության մակարդակի տարբերություն:
2. Կոնտրաստ (Contrast): Պատկերների հակադրության (contrast) տարբերություն:
3. Կառուցվածք (Structure): Պատկերների կառուցվածքային հատկությունների համեմատություն:

Այն չափում է, թե որքանով են պատկերները միմյանման իրենց կառուցվածքային առանձնահատկություններով, որոնք ավելի մոտ են մարդու տեսողական ընկալմանը:

SSIM-ը վերադարձնում է արժեք, որը գտնվում է  $-1$ -ից  $1$ -ի միջակայքում.

- $1$ : Պատկերները միանգամայն նույնն են:
- $0$ : Պատկերները չեն կիսում կառուցվածքային նմանություն:
- $< 0$ : Ամբողջովին տարբեր պատկերներ:

PSNR (Peak Signal-to-Noise Ratio)-ը պատկերների կամ տեսանյութերի որակը գնահատելու համար կիրառվող չափանիշ է: Այն չափում է, թե որքանով է սեղմված կամ վերականգնված պատկերը մոտ բուն պատկերին՝ համեմատելով նրանց պիքսելային արժեքների տարբերությունները: PSNR-ը չափվում է դեցիբելներով (dB), որտեղ բարձր արժեքները ցույց են տալիս ավելի լավ որակ: MSE-ն չափում է երկու պատկերի միջև պիքսելային տարբերությունների քառակուսային միջինը: PSNR-ը հարաբերություն է առավելագույն ուժգնության և MSE-ի միջև: Այն վերաձևվում է դեցիբելային մասշտաբի՝ օգտագործելով լոգարիթմ:

$$PSNR = 10 \times \log_{10} \times \frac{MAX_I^2}{MSE}$$

- Բարձր PSNR (օրինակ՝  $> 30$  dB): Կրկնօրինակն ու բուն պատկերը շատ նման են:
- Միջին PSNR (20-30 dB): Որակը միջին է:
- Ցածր PSNR ( $< 20$  dB): Որակը վատ է, նկատելի տարբերություններ կան:

VGG19-ն ունի 19 ընդհանուր շերտ (16 կոնվոլյուցիոնալ շերտ + 3 fully connected շերտեր): Բոլոր կոնվոլյուցիոնալ շերտերը ունեն 3x3 քանակի kernel: Յուրաքանչյուր կոնվոլյուցիոնալ բլոկի վերջում կա 2x2 չափի max pooling շերտ, որն օգտագործվում է տվյալների չափը նվազեցնելու և հաշվարկները պարզեցնելու համար: Մոդելը օգտագործում է ReLU ակտիվացման ֆունկցիա բոլոր շերտերում, ինչը նպաստում է ոչ գծայինությանը և արագացնում ուսուցումը: Կոնվոլյուցիոնալ շերտերի վերջում տվյալները հարթեցվում են և փոխանցվում երեք լրիվ կապված շերտերի: Ելքային շերտը սովորաբար softmax է՝ տվյալը n դասերի միջև հավանականություն բաշխելու համար:



## ԳԼՈՒԽ 2. ԾՐԱԳՐԻ ԻՐԱԿԱՆԱՑՈՒՄ

### 2.1 ԱԼԳՈՐԻԹՄԻ ՀԻՄՆԱԿԱՆ ՔԱՅԼԵՐԸ

NST-ի ալգորիթմի հիմնական քայլերը.

1) Նախնական պատկերների տրամադրում:

Աշխատանքը սկսվում է երկու պատկերով՝

- Բովանդակության պատկեր՝ այն պատկերը, որի հիմնական կառուցվածքը կամ բովանդակությունը ցանկանում ենք պահպանել:
- Ոճի պատկեր՝ այն պատկերը, որի գեղարվեստական առանձնահատկությունները, ոճական լուծումներն ու գունային ձևավորումը ցանկանում ենք կիրառել:

2) Կոնվոլյուցիոն նեյրոնային ցանցի օգտագործում:

NST-ն օգտագործում է նախապես սովորեցված կոնվոլյուցիոն նեյրոնային ցանց՝ բովանդակության և ոճի հատկանիշները առանձնացնելու և ներկայացնելու համար օգտագործվում է VGG19 ցանցը:

Օգտագործել ենք հետևյալ շերտերը՝

```
content_layers = ['block5_conv2']  
style_layers = ['block1_conv1', 'block2_conv1', 'block3_conv1', 'block4_conv1',  
                'block5_conv1']
```

3) Տվյալների բազայի ընտրություն

ILSVRC 2012 (ImageNet Large Scale Visual Recognition Challenge 2012) տվյալների բազայի ընտրություն, որը լայնորեն օգտագործվում է computer vision խնդիրներում: ImageNet-ն ունի ավելի քան 14 միլիոն պատկեր: Ճանաչում է 1000 class-երի օբյեկտներ, 1.2 միլիոն training նկարներ, 50.000 validation նկարներ, 100,000 test նկարներ:

4) Կորուստի ֆունկցիա:

NST-ում սահմանվում է կորուստի ֆունկցիա, որը բաղկացած է երկու հիմնական մասից՝ բովանդակության կորուստ և ոճի կորուստ: Բովանդակության կորուստը գնահատում է, թե որքանով է սինթեզված պատկերը պահպանել բովանդակության պատկերի կառուցվածքը: Ոճի

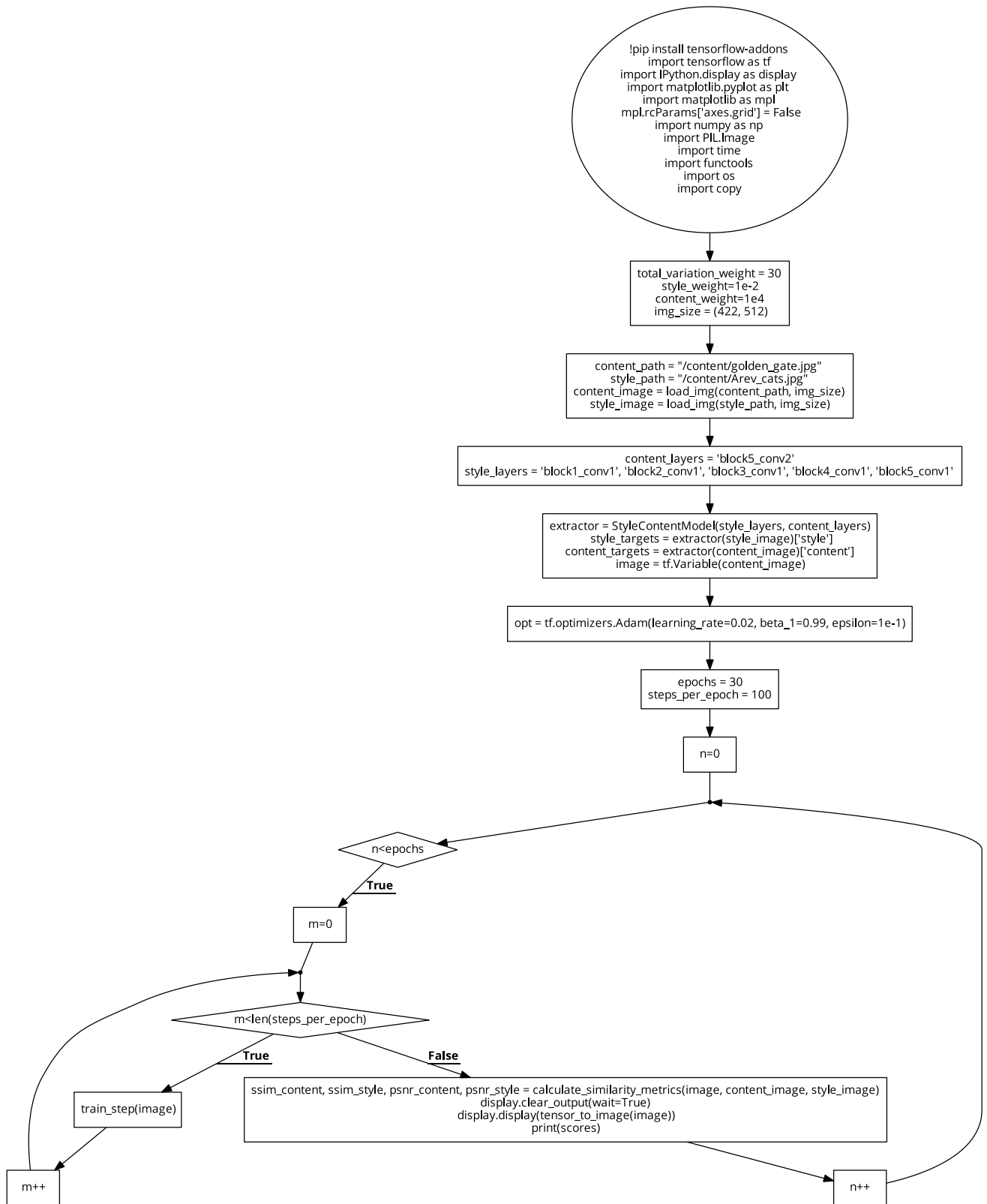
կորուստը հաշվարկում է, թե որքանով է սինթեզված պատկերը մոտ ոճային պատկերի գեղարվեստական առանձնահատկություններին: Որպես կորստի ֆունկցիա օգտագործվել է միջին քառակուսային սխալանքը:

#### 5) Training պարամետրեր

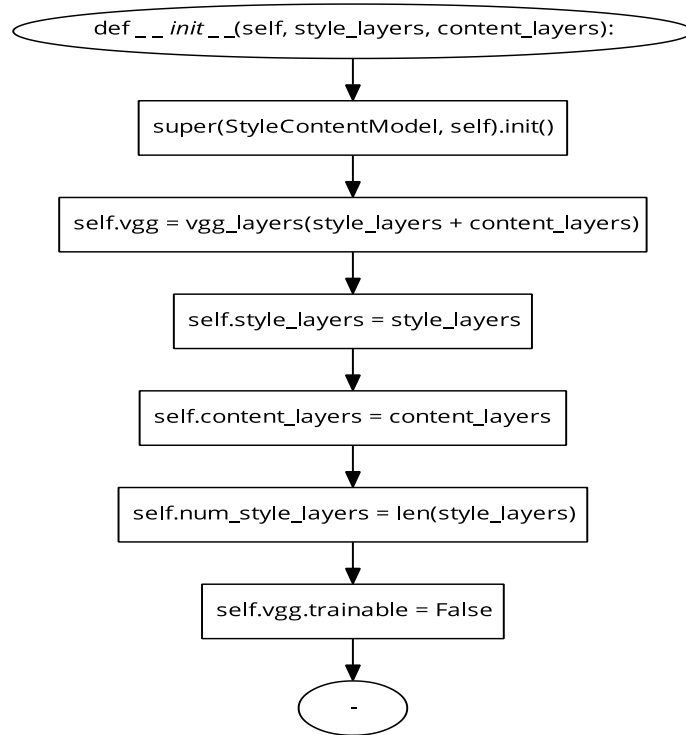
##### Հիպերպարամետրեր

- վերջնական նկարի փափկության և առանց աղմուկ աշխատանքի կշիռ  
 $total\_variation\_weight = 30$
- ոճի նկարի կշիռ  
 $style\_weight = 1e-2$
- բովանդակության նկարի կշիռ
- $content\_weight = 1e4$   
նկարի չափեր  
 $img\_size = (422, 512)$
- #Ուսուցման պարամետրեր  
 $epochs = 30$   
 $steps\_per\_epoch = 100$

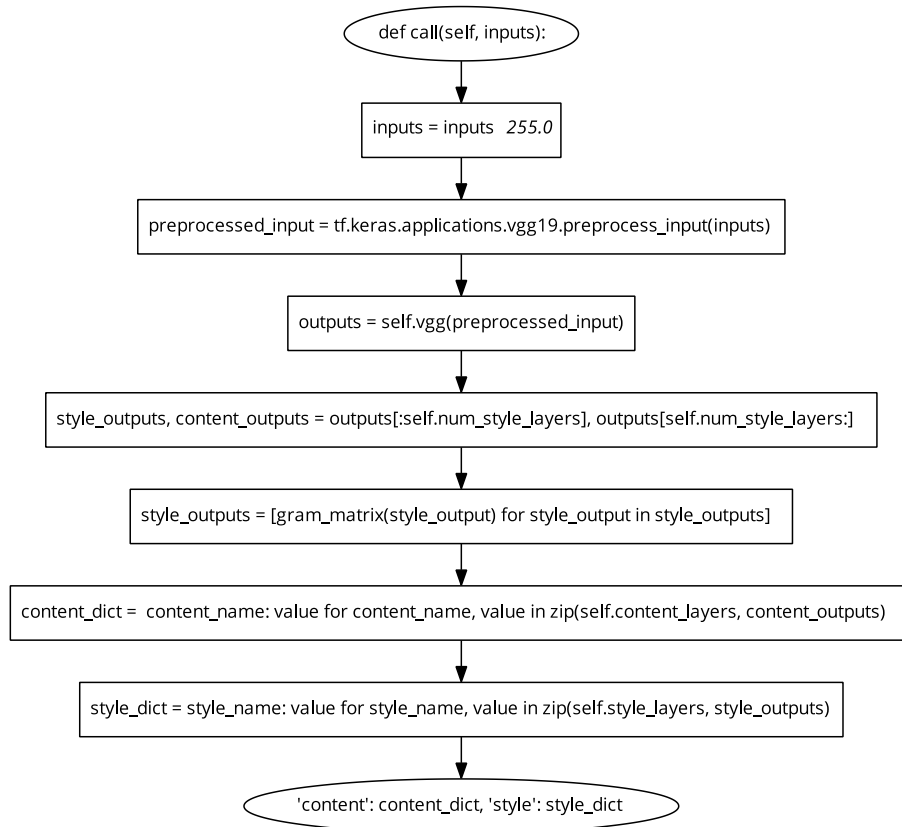
## 2.2 ԾՐԱԳՐԻ ԻՐԱԿԱՆԱՑՄԱՆ ԲԼՈԿ ՍԽԵՄԱ



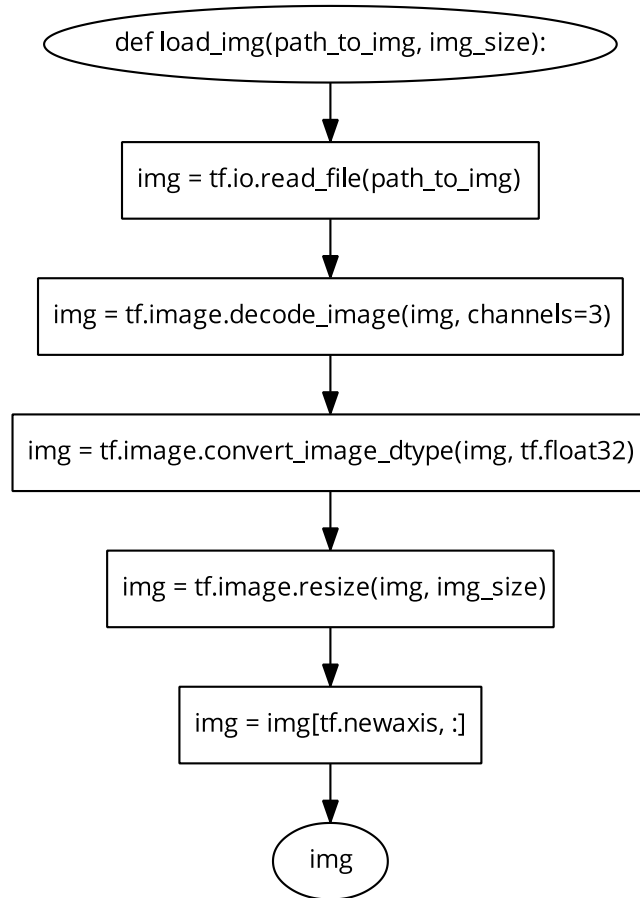
Նկ.2.1 Ծրագրի իրականացման բլոկ սխեմա



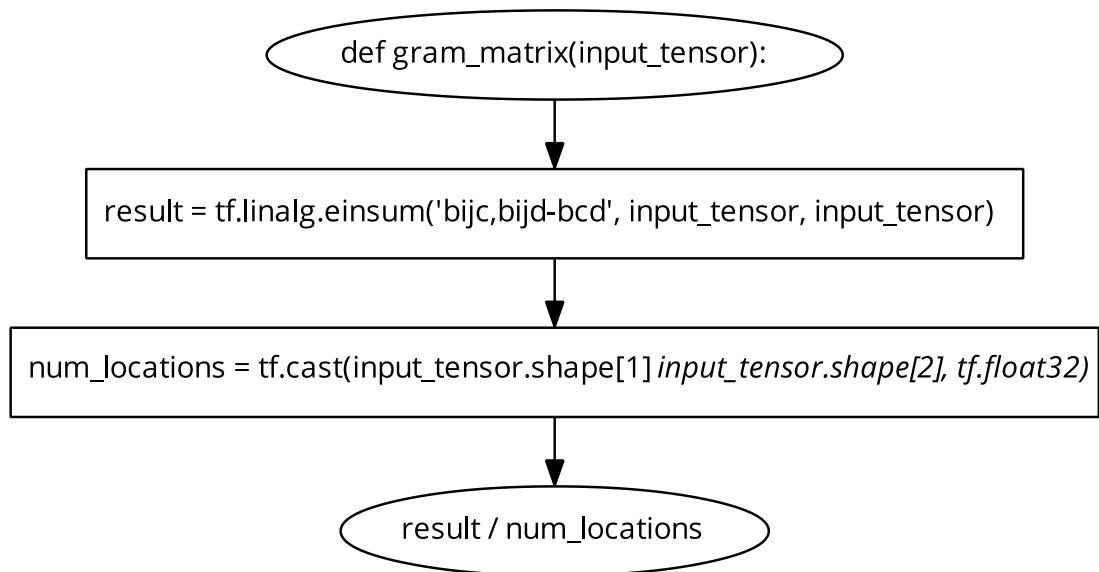
Նկ.2.2 StyleContentModel class-ի `__init__` ֆունկցիայի բլոկ սխեմա



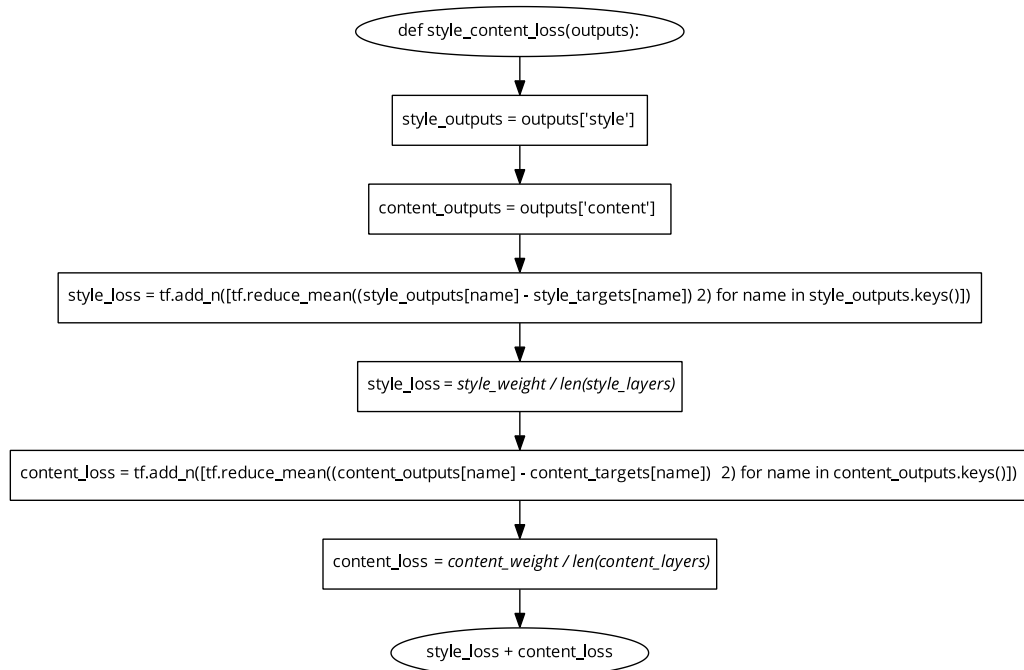
Նկ.2.3 StyleContentModel class-ի `call` ֆունկցիայի բլոկ սխեմա



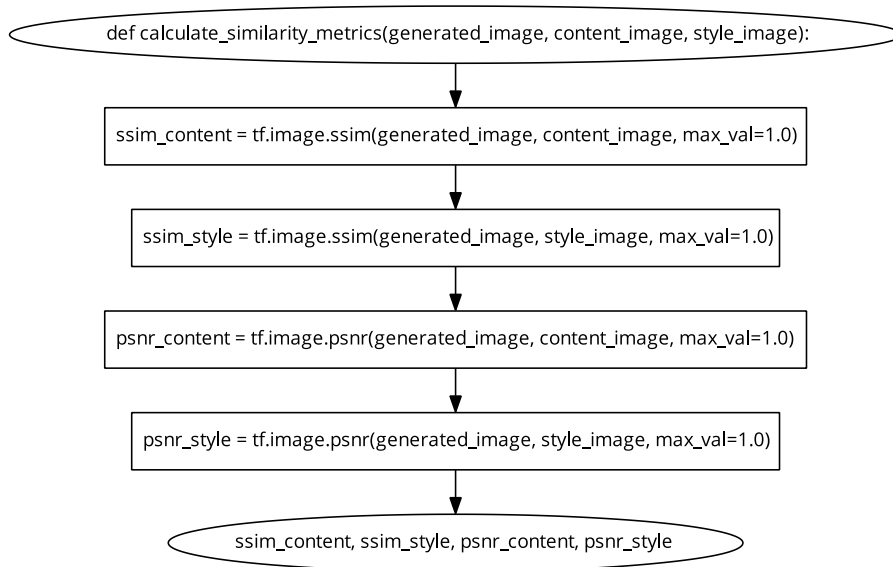
Նկ.2.4 Load\_img ֆունկցիայի բլոկ սխեմա



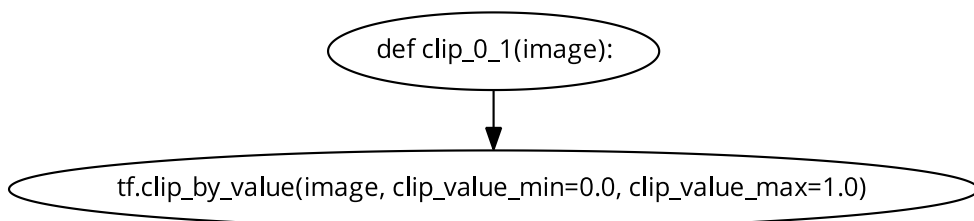
Նկ.2.5 Gram\_matrix ֆունկցիայի բլոկ սխեմա



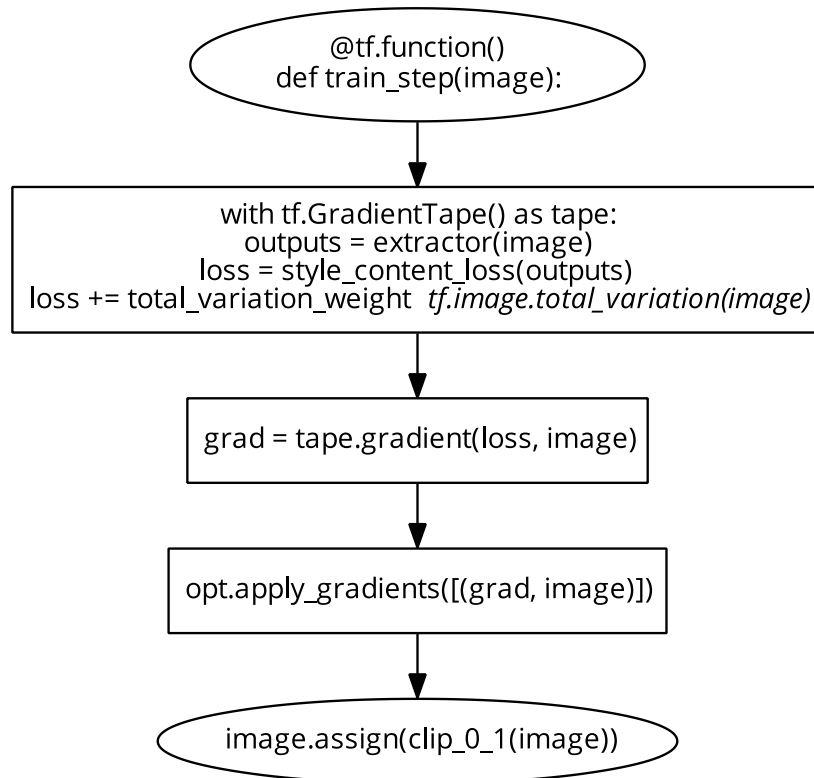
Նկ.2.6 Style\_content ֆունկցիայի բլոկ սխեմա



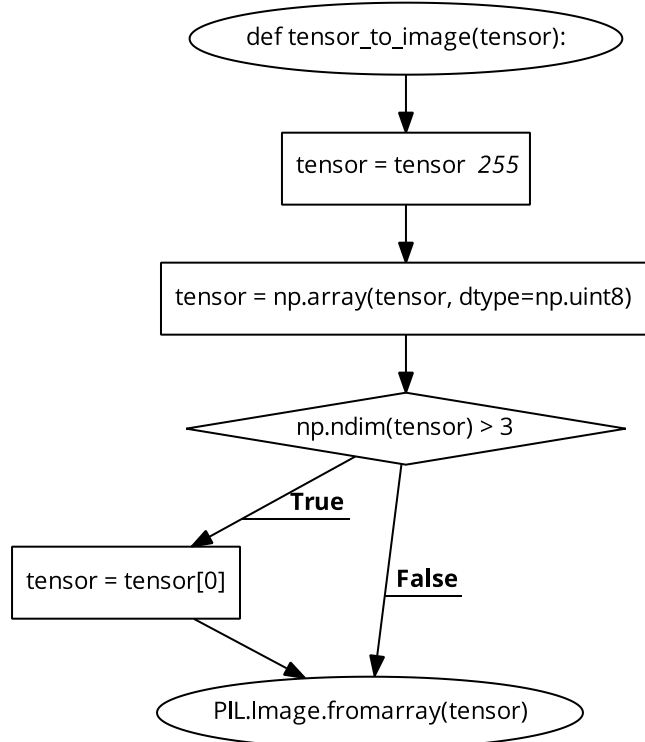
Նկ.2.7 Calculate\_similarity\_matrices ֆունկցիայի բլոկ սխեմա



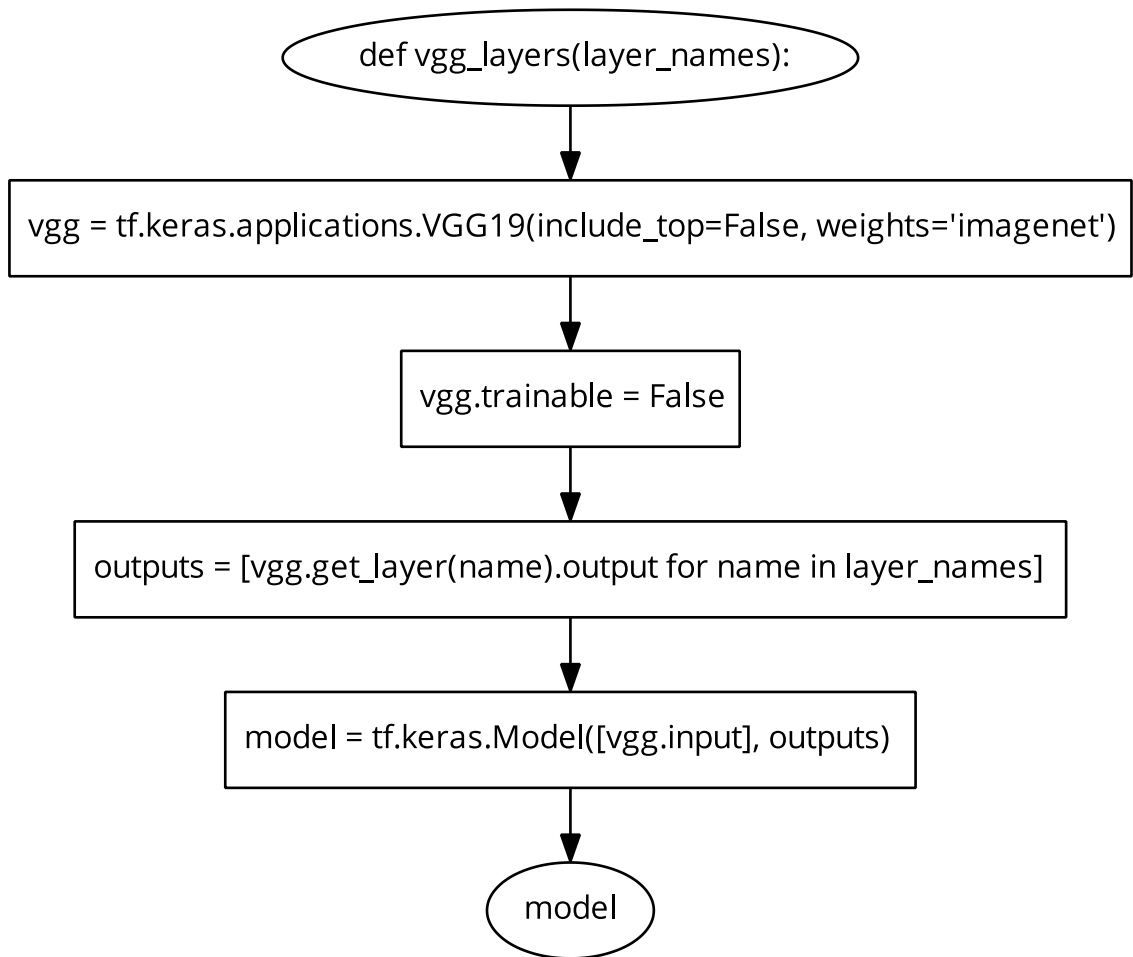
Նկ.2.8 Clip ֆունկցիայի բլոկ սխեմա



Նկ.2.9 Train\_step ֆունկցիայի բլոկ սխեմա



Նկ.2.10 Tensor\_to\_image ֆունկցիայի բլոկ սխեմա



Նկ.2.11 Vgg\_layers ֆունկցիայի բլոկ սխեմա



## ԱՐԴՅՈՒՆՔՆԵՐ



Նկ. 3.1 կառուցվածք



Նկ. 3.2 ՈՃ



Նկ. 3.3 NST-ի արդյունքը

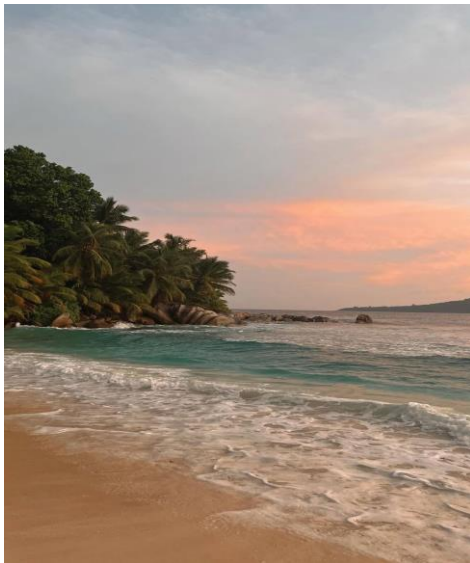
SSIM between generated and content image: [0.19573657]

SSIM between generated and style image: [0.0371879]

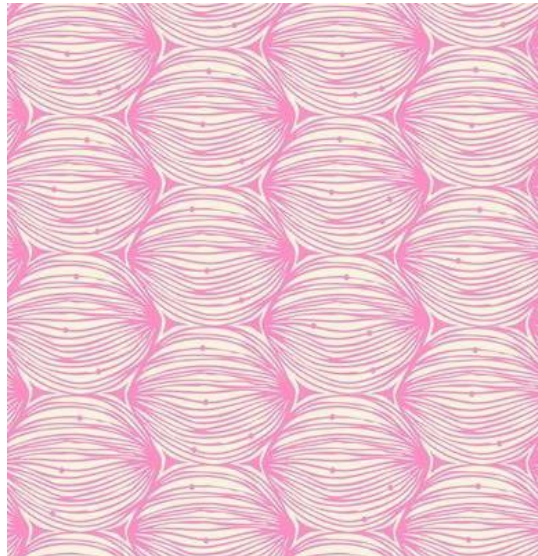
PSNR between generated and content image: [12.851691]

PSNR between generated and style image: [9.814261]

4m 32s



Նկ. 3.4 կառուցվածք



Նկ. 3.5 ՈՃ



Նկ. 3.6 NST-ի արդյունքը

SIM between generated and content image: [0.4029814]

SSIM between generated and style image: [0.12459307]

PSNR between generated and content image: [12.1038685]

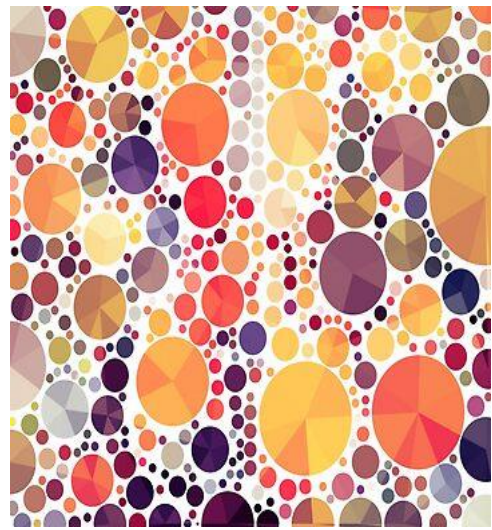
PSNR between generated and style image: [12.068717]

4m14s





Նկ. 3.7 կառուցվածք



Նկ. 3.8 ՈՃ



Նկ. 3.6 NST-ի արդյունքը

SSIM between generated and content image: [0.24694051]

SSIM between generated and style image: [0.1088256]

PSNR between generated and content image: [10.954328]

PSNR between generated and style image: [8.718956]

4m 15s

## ԵԶՐԱԿԱՑՈՒԹՅՈՒՆ

Neural Style Transfer-ի միջոցով ստացված պատկերների վերլուծությունը ցույց է տալիս արհեստական բանականության ունակությունը՝ արդյունավետորեն համադրելու բովանդակային և ոճային հատկանիշները՝ ստեղծելով նորարարական ու արտահայտիչ պատկերներ:

### 1. Բնօրինակ կառուցվածքի պահպանում:

NST-ի հիմնական առավելություններից է բովանդակային պատկերի կառուցվածքային հատկանիշների պահպանությունը: Չնայած ոճային փոփոխություններին, բովանդակային պատկերում առկա օբյեկտների հիմնական ձևերը, տեղաբաշխումն ու համատեքստը մնացել են անփոփոխ:

### 2. Ոճային ազդեցություն:

NST-ն հաջողությամբ փոխանցում է ոճային պատկերում առկա գունային շերտերը, նախշերն ու տեքստուրաները: Տարբեր ոճային պատկերներ օգտագործելով՝ հնարավոր է ստանալ զգալիորեն տարբեր արդյունքներ, ինչը վկայում է տեխնիկայի ճկունության և հարմարվողականության մասին:

### 3. Բովանդակության և ոճի ինտեգրացիա:

Ստացված պատկերները վկայում են, որ NST-ն կարող է ներդաշնակորեն համադրել բովանդակային և ոճային տարրերը: Սակայն որոշ դեպքերում ոճային ազդեցությունը գերիշխում է, ինչը կարող է հանգեցնել բովանդակության որոշ մանրամասների կորուստի:

### 4. Գունային արտահայտչականություն:

NST-ի միջոցով գունային տրանսֆորմացիաներն ակնհայտորեն բարձրացնում են պատկերների արտահայտչականությունը: Տարբեր գունային սխեմաներ և նախշեր կարող են ամբողջությամբ փոխել բնօրինակ պատկերի տրամադրությունը:

## ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ

1. <https://colab.google/>
2. <https://paperswithcode.com/>
3. <https://app.code2flow.com/>
4. Gatys et al., 2016 – Image Style Transfer Using CNNs.
5. Jing et al., 2019 – Neural Style Transfer: A Review.
6. Goodfellow et al., 2016 – Deep Learning.