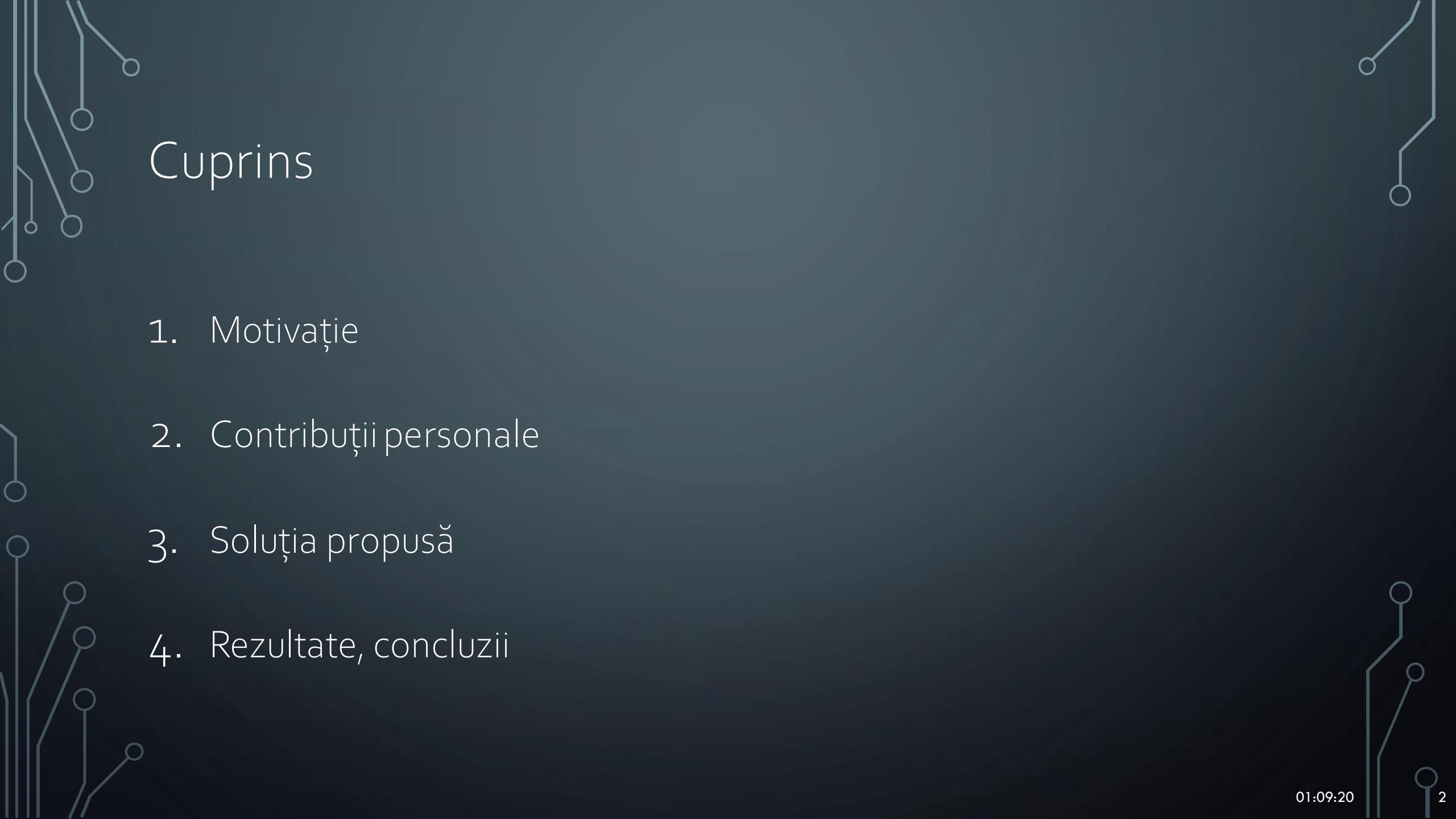




# Colorarea imaginilor alb-negru prin metode de Deep Learning

Absolvent: Maria Hlușneac

Coordonator științific: Asist. Dr. Eugen Nicolae Croitoru



# Cuprins

1. Motivație
2. Contribuții personale
3. Soluția propusă
4. Rezultate, concluzii

# 1. Motivație

- Lipsa de expresivitate a imaginilor alb-negru
- Dorință de a agrega diverse configurații de modele neuronale împreună cu rezultatele lor într-un singur loc, pentru o viziune de ansamblu mai clară

## 2. Contribuții personale

2.1. Crearea setului de date

2.2. Testarea mai multor configurații de modele neuronale

2.3. Crearea unei arhitecturi noi de rețea neuronală

## 2. Contribuții personale

### 2.1. Crearea setului de date

- Adăugare de imagini obținute de pe Youtube
- Adăugare de imagini obținute printr-un API de la Flickr
- Preluare de imagini din seturi de date de pe internet

=> Set de date portrete: 32000 de imagini preluate de pe internet

=> Set de date peisaje:

1. 3600 Youtube, 400 Flickr, 28000 de imagini de pe internet
2. 3600 Youtube, 400 Flickr, 78000 de imagini de pe internet

## 2. Contribuții personale

### 2.2. Testarea mai multor modele neuronale

- Făcând research și comparând modelele descrise în literatură, am ajuns la crearea a:
  - 4 modele de tip autoencoder
  - 15 modele de tip GAN

## 2. Contribuții personale

### 2.3. Crearea unei arhitecturi noi de rețea neuronală

- Detalii la prezentarea soluției propuse

### 3. Soluția propusă: colorarea imaginilor de tip **portret și peisaj**

- Scop: o colorare cât mai plauzibilă
- Modul de realizare:
  - rețea de predicție (colorare) – 2 abordări cu modele neuronale

# Parcursul unei imagini de la alb-negru la culoare

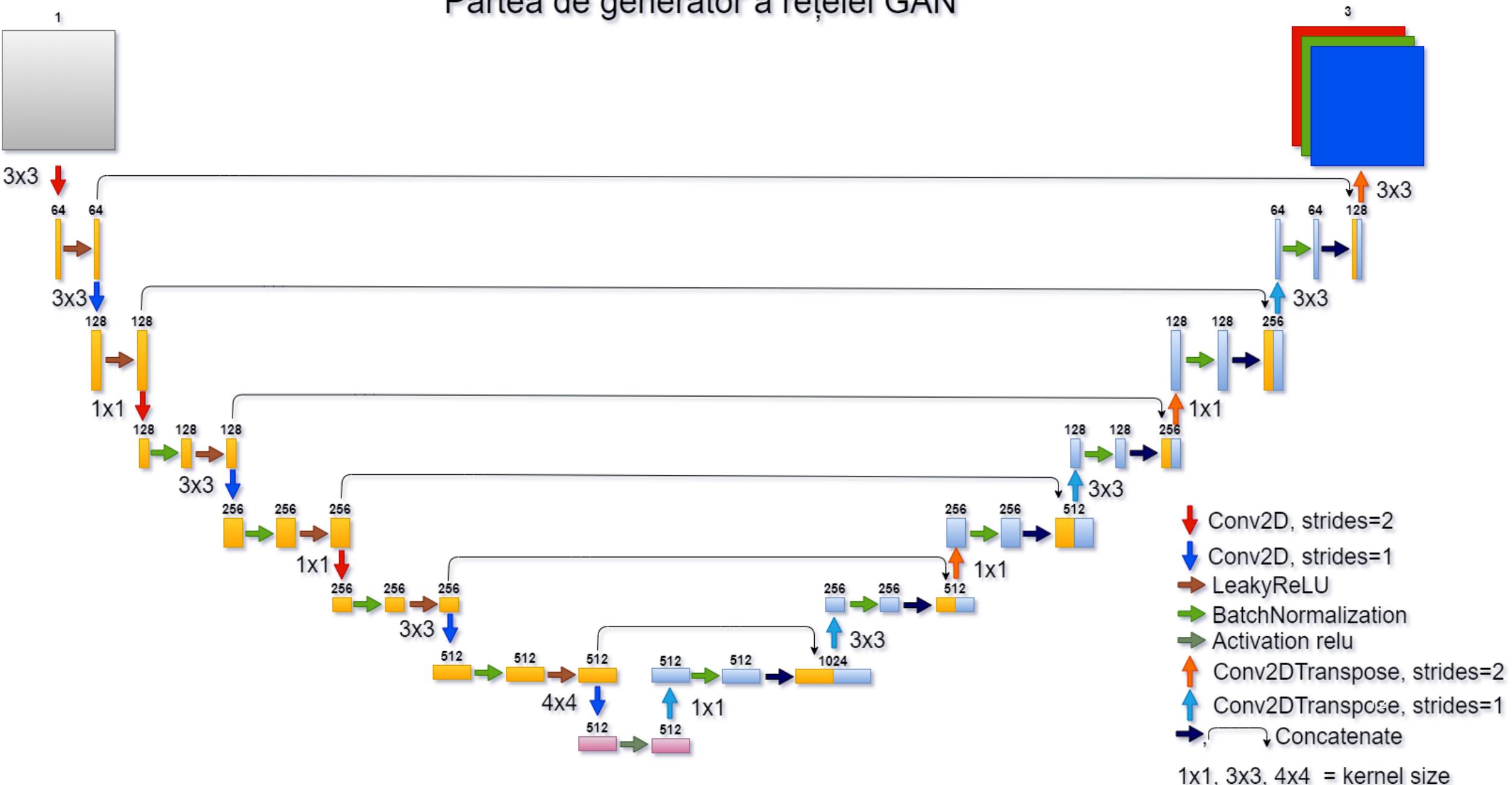


## Rețeaua de predicție (colorare)

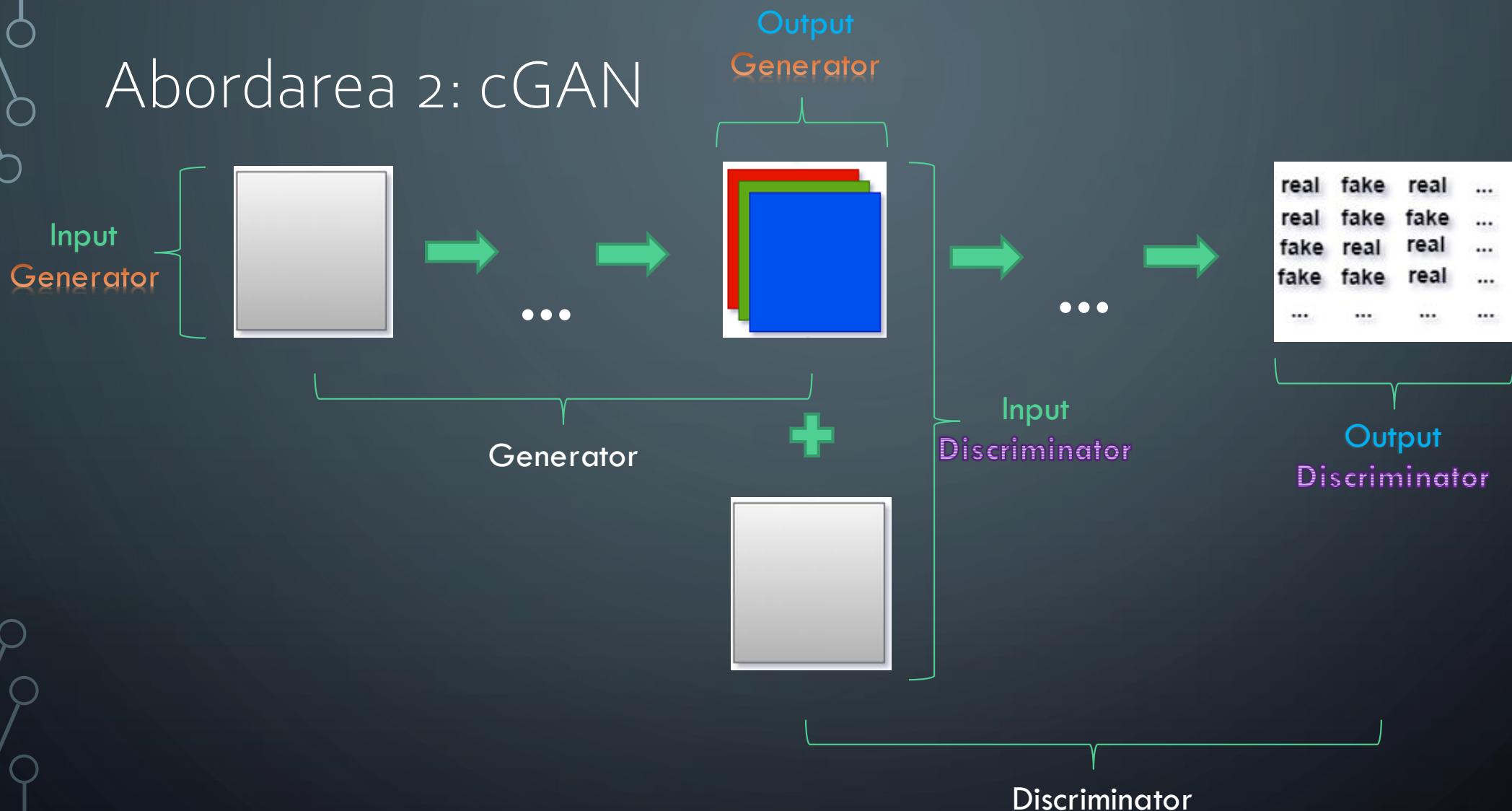
# Abordarea 1: Autoencoder

- 4 configurații:
  - U-Net + metodă de upsampling de tip învățare
  - U-Net + metodă de upsampling de tip interpolare
  - fără U-Net + metodă de upsampling de tip învățare
  - fără U-Net + metodă de upsampling de tip interpolare

# Autoencoder de tip U-Net cu upsampling prin metoda Conv2DTranspose / Partea de generator a rețelei GAN



## Abordarea 2: cGAN



## Abordarea 2: cGAN (Conditional GAN)

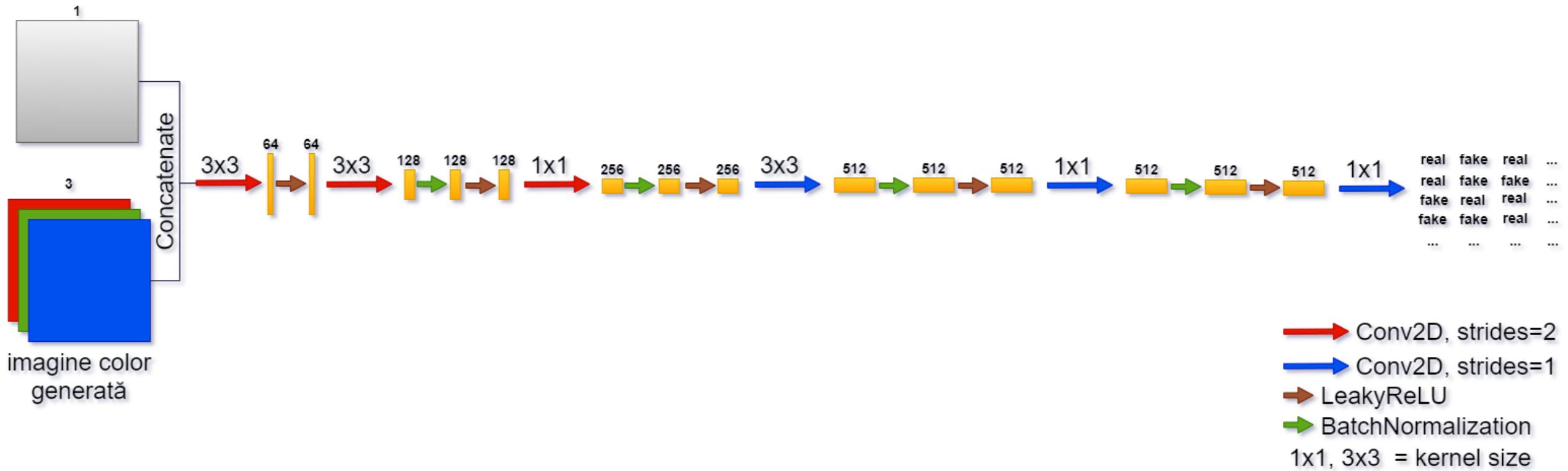
- 15 configurații:
  - 4 rețele de tip generator, 3 rețele de tip discriminator  
=> 12 modele GAN
  - 3 rețele ce provin din cel mai bun model selectat din cele 12

# Abordarea 2: cGAN

## Generatorul și discriminatorul

- Generatorul ~ autoencoder-ul de la Abordarea 1
- Discriminatorul este de tip PatchGAN și a fost implementat în 2 moduri:
  1. aplicat peste toată imaginea
  2. convoluționat peste porțiuni de imagine, urmând ca rezultatele de după convoluționarea fiecărei porțiuni să fie concatenate pentru a construi output-ul
    - 2.1. convoluționat peste  $4 \times 4$  porțiuni de imagine
    - 2.2. convoluționat peste  $8 \times 8$  porțiuni de imagine

## Discriminator de tip PatchGAN





Alb-negru



Generat



Original



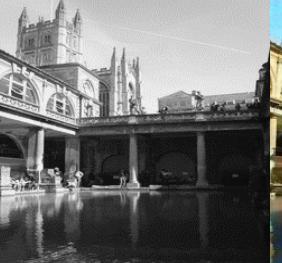
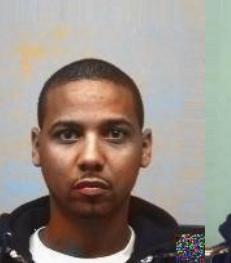
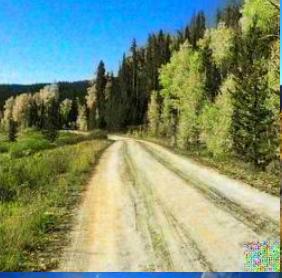
Alb-negru



Generat



Original



# Autoencoder - Portrete

U-Net, învățare



U-Net, interpolare



Original



nu U-Net, învățare



nu U-Net, interpolare



Original



# Autoencoder - Portrete



U-Net, interpolare



Original



nu U-Net, învățare



nu U-Net, interpolare

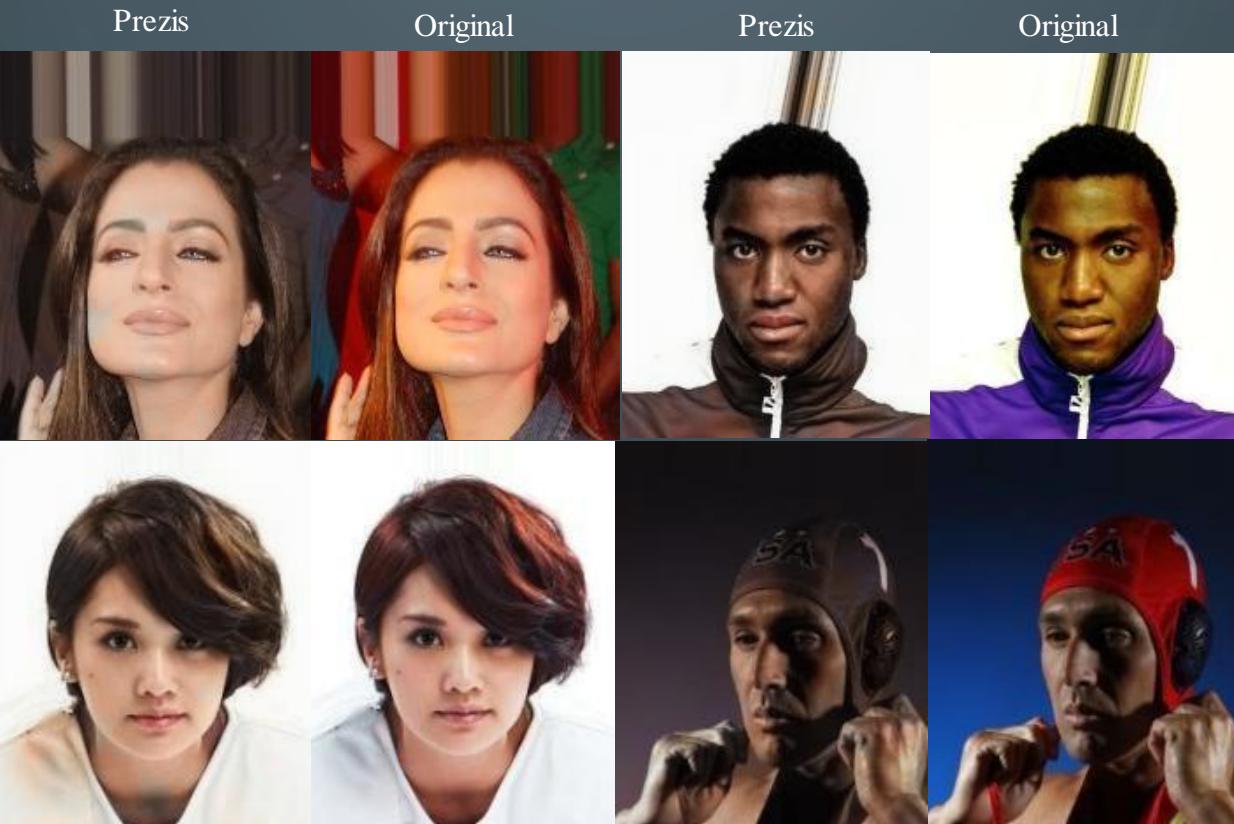


Original



# Autoencoder – Portrete

## Rezultate mai puțin satisfăcătoare



# Autoencoder - Peisaje

70 de epoci, 32 000 de imagini,  
U-Net, învățată

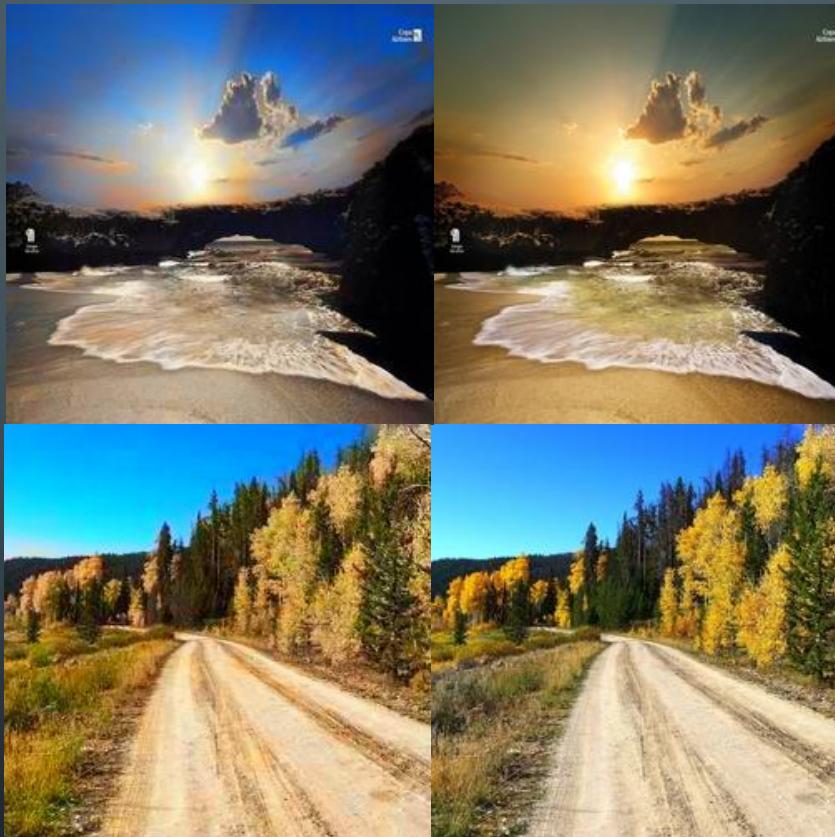
30 de epoci, 82 000 de imagini,  
U-Net, învățată

Original



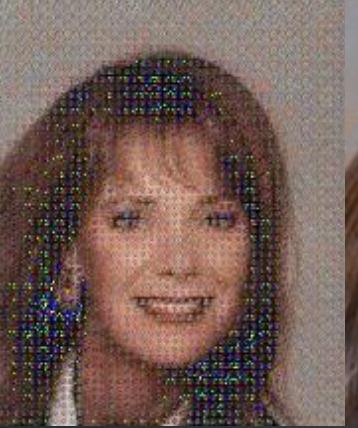
# Autoencoder – Peisaje Postprocesare

Procesat



# GAN - Portrete

U,  $\hat{\text{inv}}$  dO, lw=[20, 80]



U,  $\hat{\text{inv}}$ , d4, lw=[20, 80]



Original

U,  $\hat{\text{inv}}$ , d8, lw=[1, 99]



U,  $\hat{\text{inv}}$ , d8, lw=[35, 65]



Original

Original

nU, int, dO, lw=[20, 80]



U,  $\hat{\text{inv}}$ , d8, lw=[20, 80]



Original

nU,  $\hat{\text{inv}}$ , d4, lw=[20, 80]



nU,  $\hat{\text{inv}}$ , d8, lw=[20, 80]

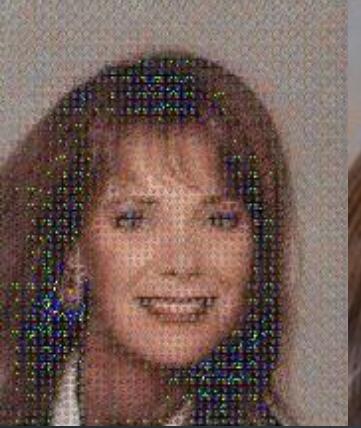


Original

Original

# GAN - Portrete

U,  $\hat{\text{inv}}$  dO, lw=[20, 80]



U,  $\hat{\text{inv}}$ , d4, lw=[20, 80]



Original

U,  $\hat{\text{inv}}$ , d8, lw=[1, 99]



U,  $\hat{\text{inv}}$ , d8, lw=[35, 65]



Original

Original

nU, int, dO, lw=[20, 80]



U,  $\hat{\text{inv}}$ , d8, lw=[20, 80]



Original

nU,  $\hat{\text{inv}}$ , d4, lw=[20, 80]



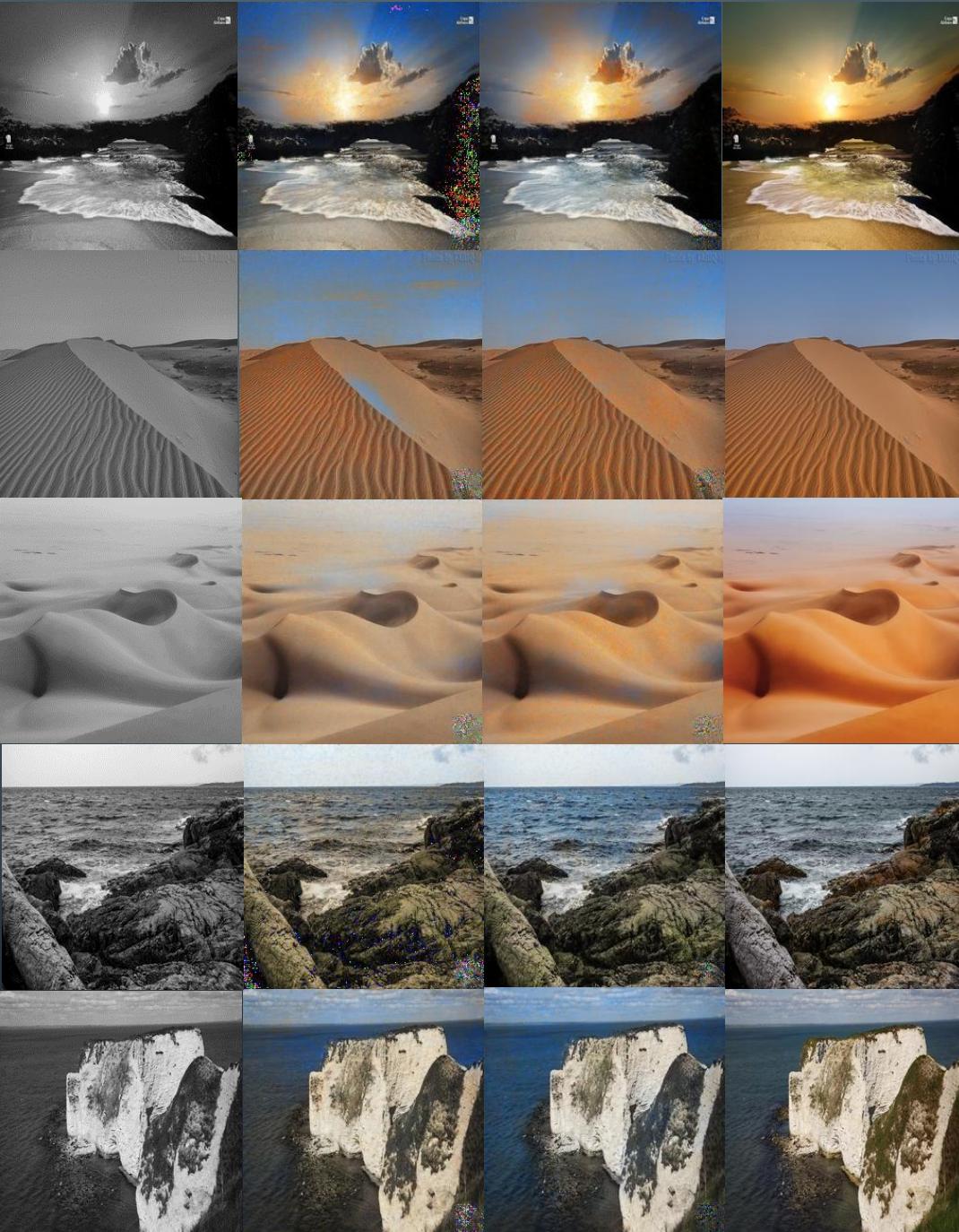
nU,  $\hat{\text{inv}}$ , d8, lw=[20, 80]



Original

# GAN – Peisaje

Alb-negru    32000imagini,30ep    82000imagini,15ep    Original



# Autoencoder vs GAN

- Metrica folosită pentru cuantificarea rezultatelor:

Diferența medie absolută (per pixel) în medie

(diferența este calculată între valorile pixelilor imaginii target și cele ale imaginii prezise)

- $0 \leq \text{valoare metrică} \leq 1$
- Lipsa de expresivitate a metricii

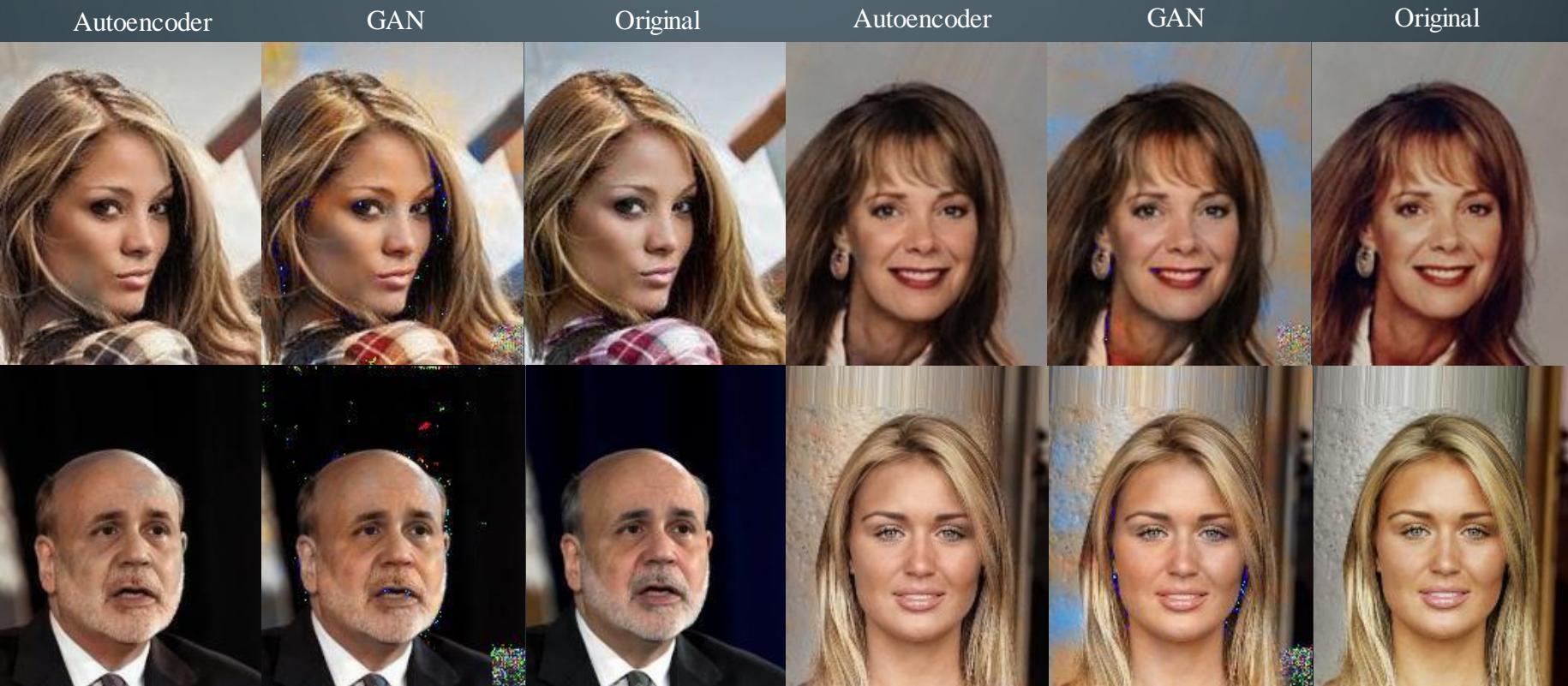
# Autoencoder vs GAN – Portrete

## Rezultate pentru modelele cele mai bune

	Autoencoder	GAN
Diferența medie minimă	0,009	0,011
Diferența medie în medie	0,037	0,042
Diferența medie maximă	0,161	0,173
Număr de epoci	100	60
Set de date	32768 imagini antrenament, 3072 imagini test	

Conform metricii, autoencoder-ul este mai bun decât GAN-ul.

# Autoencoder vs GAN – Portrete Exemplu care contrazic metrică



# Autoencoder vs GAN – Peisaje

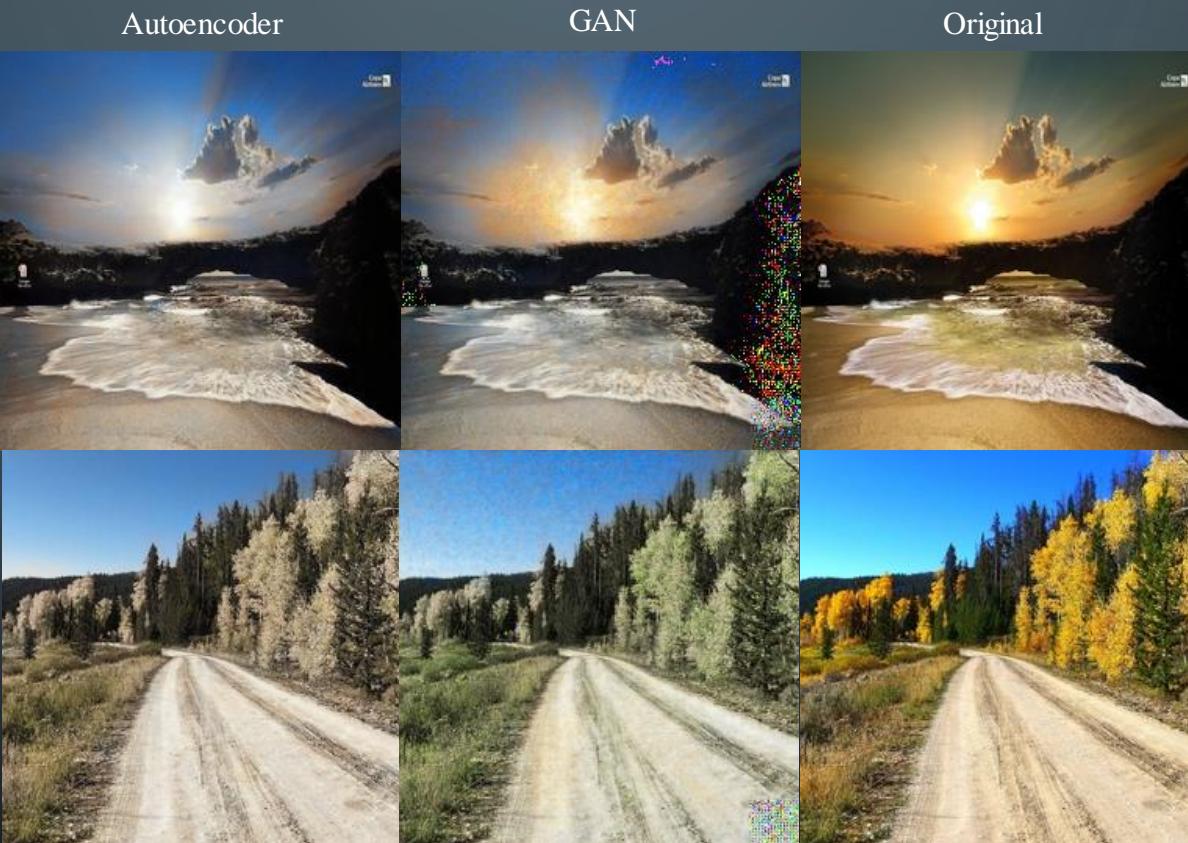
## Rezultate pentru modelele cele mai bune

	AE - set restrâns	G - set restrâns	AE - set extins	G – set extins
Diferență medie minimă	0,021	0,029	0,019	0,024
Diferență medie în medie	0,086	0,087	0,083	0,159
Diferență medie maximă	0,443	0,415	0,459	0,456
Număr epoci	70	30	30	15
Set de date	32768 antrenament, 4096 test		81920 antrenament, 8192 test	

Conform metricii, și de data aceasta autoencoder-ul s-a descurcat mai bine decât GAN-ul.

# Autoencoder vs GAN – Peisaje

## Exemplu care contrazic metrică



## Viziune de ansamblu

- Ambele modele dă rezultate relativ plauzibile
- Autoencoder-ul și GAN-ul s-au descurcat mai bine pe portrete decât pe peisaje
- Autoencoder-ul are un comportament mai “rezervat”, GAN-ul “riscă” uneori prea mult
- Unele rezultate mai puțin satisfăcătoare pot fi ajustate cu o scurtă procesare
- Autoencoder-ul câștigă la calcule, GAN-ul câștigă din punct de vedere vizual (opinie personală)

## Concluzii

- ✓ colorarea plauzibilă a imaginilor de tip peisaj și portret
- ✓ comparație între mai multe configurații de modele neuronale
- Mai sunt multe de explorat (CycleGAN, ProGAN, colorare de videoclipuri)

# Demo

Vă mulțumesc  
pentru atenție!