Outline

- Artist-Critic Co-Evolution
- Co-Evolution Paradigms
- Blind Watchmaker (GP Artist, Human Critic)
- Evolutionary Art (GP Artist, GP or NN Critic)
- Generative Adversarial Networks (CNN Artist, CNN Critic)
Critic is rewarded for distinguishing real images from those generated by the artist.

Artist is rewarded for fooling the critic into thinking that generated images are real.
## Co-Evolution Paradigms

<table>
<thead>
<tr>
<th>Artist</th>
<th>Critic</th>
<th>Method</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biomorph GP &lt;br&gt;CPPN CA</td>
<td>Human &lt;br&gt;Human</td>
<td>Blind Watchmaker &lt;br&gt;PicBreeder &lt;br&gt;EvoEco</td>
<td>(Dawkins, 1986) &lt;br&gt;(Sims, 1991) &lt;br&gt;(Secretan, 2011) &lt;br&gt;(Kowaliw, 2012)</td>
</tr>
<tr>
<td>GP Photo Agents GP</td>
<td>SOM NN &lt;br&gt;NN NN</td>
<td>Artificial Creativity &lt;br&gt;Computational Aesthetics &lt;br&gt;Evolutionary Art &lt;br&gt;Aesthetic Learning</td>
<td>(Saunders, 2001) &lt;br&gt;(Datta, 2006) &lt;br&gt;(Machado, 2008) &lt;br&gt;(Greenfield, 2009) &lt;br&gt;(Li &amp; Hu, 2010)</td>
</tr>
<tr>
<td>HERCL HERCL</td>
<td>HERCL DCNN</td>
<td>Co-Evolving Line Drawings &lt;br&gt;HERCL Function/CNN</td>
<td>(Vickers, 2017) &lt;br&gt;(Soderlund, 2018)</td>
</tr>
<tr>
<td>DCNN DCNN</td>
<td>DCNN DCNN</td>
<td>Generative Adversarial Nets &lt;br&gt;Plug &amp; Play Generative Nets</td>
<td>(Goodfellow, 2014) &lt;br&gt;(Nguyen, 2016)</td>
</tr>
</tbody>
</table>
the Human is presented with 15 images

the chosen individual is used to breed the next generation
Blind Watchmaker Biomorphs

- Swallowtail
- Man in hat
- Lunar lander
- Precision balance
- Caddis
- Scorpion
- Cat's cradle
- Tree frog
- Spitfire
- Crossed sabres
- Bee-flower
- Shelled cephalopod
- Insect
- Fox
- Lamp
- Jumping Spider
- Bat
Blind Watchmaker (Sims, 1991)

- **Artist** = Genetic Program (GP)
  - used as function to compute R,G,B values for each pixel $x, y$
- **Critic** = Human
PicBreeder (Secretan, 2011)

- Artist = Convolutional Pattern Producing Neural Network (CPPN)
- Critic = Human
- interactive Web site (picbreeder.org) where you can choose existing individual and use it for further breeding

- Blind Watchmaker paradigm is cool, but it may require a lot of work from the Human
- Can the Human be replaced by an automated Critic?
## Image Generating Paradigms

<table>
<thead>
<tr>
<th>Biomorph</th>
<th>GP</th>
<th>Picbreeder</th>
<th>CA</th>
</tr>
</thead>
<tbody>
<tr>
<td>HERCL(draw)</td>
<td>HERCL(func)</td>
<td>HERCL(func)</td>
<td>GAN</td>
</tr>
</tbody>
</table>

Biomorph, GP, Picbreeder, and CA are methods for generating images. HERCL (Hullermeier Evolutionary Rule Learning) is used in both the draw and function modes.

GAN (Generative Adversarial Networks) is another method for generating images.
Generative Adversarial Networks

Generator (Artist) $G_\theta$ and Discriminator (Critic) $D_\psi$ are both Deep Convolutional Neural Networks.

Generator $G_\theta : z \mapsto x$, with parameters $\theta$, generates an image $x$ from latent variables $z$ (sampled from a Normal distribution).

Discriminator $D_\psi : x \mapsto D_\psi(x) \in (0, 1)$, with parameters $\psi$, takes an image $x$ and estimates the probability of the image being real.

Generator and Discriminator play a 2-player zero-sum game to compute:

$$\min_{\theta} \max_{\psi} \left( \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log D_\psi(x) \right] + \mathbb{E}_{z \sim p_{\text{model}}} \left[ \log (1 - D_\psi(G_\theta(z))) \right] \right)$$

Discriminator tries to maximize the bracketed expression, Generator tries to minimize it.
Generative Adversarial Networks

Alternate between:

Gradient ascent on Discriminator:

$$\max_{\psi} \left( \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log D_\psi(x) \right] + \mathbb{E}_{z \sim p_{\text{model}}} \left[ \log(1 - D_\psi(G_\theta(z))) \right] \right)$$

Gradient descent on Generator, using:

$$\min_{\theta} \mathbb{E}_{z \sim p_{\text{model}}} \left[ \log(1 - D_\psi(G_\theta(z))) \right]$$
Generative Adversarial Networks

Alternate between:

Gradient ascent on Discriminator:

$$\max_{\psi} \left( E_{x \sim p_{data}} \left[ \log D_{\psi}(x) \right] + E_{z \sim p_{model}} \left[ \log (1 - D_{\psi}(G_{\theta}(z))) \right] \right)$$

Gradient descent on Generator, using:

$$\min_{\theta} E_{z \sim p_{model}} \left[ \log (1 - D_{\psi}(G_{\theta}(z))) \right]$$

This formula puts too much emphasis on images that are correctly classified. Better to do gradient ascent on Generator, using:

$$\max_{\theta} E_{z \sim p_{model}} \left[ \log (D_{\psi}(G_{\theta}(z))) \right]$$

This puts more emphasis on the images that are wrongly classified.
GAN properties:

- one network aims to produce the full range of images $x$, with different values for the latent variables $z$
- differentials are backpropagated through the Discriminator network and into the Generator network
- compared to previous approaches, the images produced are much more realistic!
Generative Adversarial Networks

repeat:

for k steps do

sample minibatch of \( m \) latent samples \( \{z^{(1)}, \ldots, z^{(m)}\} \) from \( p(z) \)
sample minibatch of \( m \) training items \( \{x^{(1)}, \ldots, x^{(m)}\} \)
update Discriminator by gradient ascent on \( \psi \):

\[
\nabla_{\psi} \frac{1}{m} \sum_{i=1}^{m} \left[ \log D_{\psi}(x^{(i)}) + \log (1 - D_{\psi}(G_{\theta}(z^{(i)}))) \right]
\]

end for

sample minibatch of \( m \) latent samples \( \{z^{(1)}, \ldots, z^{(m)}\} \) from \( p(z) \)
update Generator by gradient ascent on \( \theta \):

\[
\nabla_{\theta} \frac{1}{m} \sum_{i=1}^{m} \log(D_{\psi}(G_{\theta}(z^{(i)})))
\]

end repeat
GAN Convolutional Architectures

- normalize images to between $-1$ and $+1$
- replace pooling layers with:
  - strided convolutions (Discriminator)
  - fractional-strided convolutions (Generator)
- use BatchNorm in both Generator and Discriminator
- remove fully connected hidden layers for deeper architectures
- use tanh at output layer of Generator,
  ReLU activation in all other layers
- use LeakyReLU activation for all layers of Discriminator
Generator Architecture

Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (Radford et al., 2016)
GAN Image Vector Arithmetic

- smiling woman
- neutral woman
- neutral man
- smiling man
GAN Image Vector Arithmetic

\[
\text{man with glasses} - \text{man without glasses} + \text{woman without glasses} = \text{woman with glasses}
\]
Mediocre Stable States

- Like any coevolution, GANs can sometimes oscillate or get stuck in a mediocre stable state.
  - **oscillation**: GAN trains for a long time, generating a variety of images, but quality fails to improve (compare IPD)
  - **mode collapse**: Generator produces only a small subset of the desired range of images, or converges to a single image (with minor variations)

Methods for avoiding mode collapse:
- Conditioning Augmentation
- Minibatch Features (Fitness Sharing)
- Unrolled GANs
The GAN Zoo

- Contex-Encoder for Image Inpainting
- Texture Synthesis with Patch-based GAN
- Conditional GAN
- Text-to-Image Synthesis
- StackGAN
- Patch-based Discriminator
- $S^2$-GAN
- Style-GAN
- Plug-and-Play Generative Networks
GAN References


http://www.iangoodfellow.com/slides/2016-12-04-NIPS.pdf

https://arxiv.org/abs/1612.00005