Generating Videos With Recurrent GANs

Asmitha Rathis
Vignesh Gokul
Siddarth Ravichandran
Srihari Veeraraaghavan
Motivation

- Understand object motions and scene dynamics in videos
- Learn how scenes transform using unlabeled videos
- Can the model learn useful features that can be used in video related tasks like Action Recognition?
- Strong Representation Learning (without supervision)
“Most of human and animal learning is unsupervised learning. If intelligence was a cake, unsupervised learning would be the cake, supervised learning would be the icing on the cake, and reinforcement learning would be the cherry on the cake. We know how to make the icing and the cherry, but we do not know how to make the cake. We need to solve the unsupervised learning problem before we can even think of getting to true AI.” - Yann Lecun

“You cannot predict what you cannot understand” - Anonymous
Related Work - VGAN (Carl Vondrick et. al)
Related Work - TGAN (Masaki Saito et. al)
Related Work - MoCo GAN (Sergey Tulyakov et. al)
Limitations of Existing Models

1. Can generate videos only up to a second long
2. Assumes fixed background
Objectives

1. Generate videos for variable time periods
2. Generate videos with variable background
3. Potential use in tasks like video classification and activity recognition
Architecture
Recurrent Model

\[ \text{inpt}_t = \begin{cases} W_x f_t^{(i)} + c_t^{(i+1)}, & 1 < i < K \\ f_t^{(i=K)}, & i = K \end{cases} \]
Latent space - Intuition for Tiers

- Latent space over hierarchical GRUs to capture long term dependencies
- Each tier is a deep RNN which summarizes the history of its inputs into a conditioning vector for the next module downward
- Operates at different clock rates
Generator

- 5 layer convolution network with fractional strides.
- Batch Normalization and ReLU activation for hidden layers
- Generates 96 x 96 x 3
Discriminator - Image & Video

- 5 layer Convolutional network for Image Frames
- 5 layer 3D Convolution Network for Video
- Batch Normalization and ReLU activation for hidden layers
Training

- Loss Function

$$\max_{G,R,M} \min_{D_I,D_V} \mathbb{E}_x [-\log D_I(x)] + \mathbb{E}_{\hat{x}}[-\log(1 - D_I(\hat{x}))] + \mathbb{E}_x [-\log D_V(x)] + \mathbb{E}_{\hat{x}}[-\log(1 - D_V(\hat{x}))]$$

- Trained for 2000 epochs (~15 hours on 1070 8GB)
- Adam optimizer for weight updation
**Dataset**

- EPIC-KITCHENS 2018: First person videos of kitchen activities
- Each video is around 1 hour long
- Preprocessing:
  - Extract 1 minute videos (3600 frames)
  - Each frame is resized to 96x96x3
Results (Real Video)
Results (Preprocessed Real Video)
Results (Generated Video)
Future Goals

- Visualize the hierarchical embeddings
- Check the robustness of representations learnt
- Generate videos for longer duration (~15 min)
- Needs huge amount of resources
Questions?