A Study on the Paper: Unsupervised Doodling and Painting with Improved SPIRAL

Ege Dinçer

Koc University, Dept. of Computer Engineering

Istanbul, Turkey

# Introduction

It’s an easy task for humans to construct complicated images since they are capable to interpret the scenes and object with reasoning abilities. However, from the artificial intelligence point of view it is not a simple objective to interpret the images, which are not labeled through the training architecture. One possible solution for this problem is that using generative modeling approach.

The authors tried to reconstruct the images like a human imitates an image with a brush and a pen by creating generative models that uses physical grounding. They have trained the agents, which use same tools a human use to imitate an image such as a brush or a pen, with reinforcement learning to make them utilize the digital painting environments.

# Formal Problem Definition

Through the reinforcement learning part of the problem agents act on the simulated canvas by placing the brush strokes. For the problem, the size of the brush, the pressure to be applied on the canvas and the color chosen to make up the state space. Additionally, the state space contains the canvas, which is partially completed that includes the drawn locations in the environment, as well as the current brush location in order for the agent to comprehend the current environment. This state space is passed through the policy network in order to decide the next state which will be explained in the following parts. The locations are introduced as fixed quadratic Beziér form of coordinates that are start, mid-point, and end-coordinates of a stroke.

![A screenshot of a video game

Description automatically generated with low confidence]()

*Figure 1: Visualization of the policy network taking the states as input*

For the actions determined by agent have been specified as rendering by the authors. As they have stated agents’ actions are changing the brush size, changing the pressure they have apply on the canvas, the strokes they make on the canvas and also changing the color of the brush. The authors called the whole process of making these actions as rendering throughout the paper. At each step of the strokes in the environment, agent produces new values for the mid-point and end-coordinates as well as selecting the color of the stroke and the thickness of it. From the previous steps ending point becomes the next steps start coordinate. The paper refers this as simple interface. Authors also mentions a compound interface which explained as a course of strokes built up in the environment.

Throughout the steps in the drawing, agent produces the coordinates which are named as the control points. In order to terminate the action of the stroke, agent executes a discrete action which is the point of the stroke that is stored onto the canvas as a cubic spline going through the control points. The selection of colors of stroke and thickness of the brush have been determined before executing the stroke sequences. They have also mentioned that with this interface agents are able to produce smoother, more precise therefore more pleasing results.

A close up of a logo

Description automatically generatedA close up of a map

Description automatically generated

*Figure 2: Visualization of Action Space*

One of main parts that the authors have improved from the original SPIRAL (Synthesizing Programs for Images using Reinforced Adversarial Learning) is the rewards received by the agents. They have specified the rewards of the agents through the jointly-trained adversarial discriminator networks so the structure of it isn’t within the agent itself but it’s mostly in the environment, which the agent is in contact with. The original SPIRAL rewarded the agents only on the last step of the episode depending on the discriminator loss, however SPIRAL++ changed this feature. In the new architecture discriminator loss is calculated at every step depending on the partially drawn canvas, also at every timestep agent receives 1-step improvement in loss as its reward:

They have named this rewarding process as Temporal Credit Assignment. This current reward distribution leads to optimal policies if the discount factor γ is set to 1. Even though this causes all terms to be canceled out as stated in the paper, authors have got the best results by setting γ in between 0 to 0.99 that causes agent to become more greedy with some bias.refers the discriminator network and it calculates the loss according to the Dirac δ-function. As it’s described in the original paper R is a regularization term to constrain to stay in the set of Lipschitz continuous functions. Additionally, in the original paper Wasserstein distance was used as a loss function to feed the rewards however, SPIRAL++ has changed it to use the original minimax objective loss function to feed the reward in.

# Approach

For the reinforcement part of the problem they have used the network, which they have called as the generator network. Authors have been following the same approach for the optimal policy calculation and reward calculation except the differences mentioned in the previous section. In the original paper of SPIRAL, to utilize the maximized expected return of the

*Figure 3: Advantage Actor-critic Algorithm*

agent they have employed a variant of REINFORCE algorithm, and advantage actor-critic (A2C) have been used which is given in the *Figure 3.* A value approximation function, which is indicated as , has been used that is independent from the θ. For the reward calculation Monte-Carlo estimate of the returns have been used in the problem.

Actors in the network are responsible to produce training trajectories with interaction between the policy network and the rendering simulator. Every trajectory contains a sequence with the intermediate renderings made. To provide the optimal policy, the network receives these trajectories from the actors and combines them into a batch, after that it updates the by performing Stochastic Gradient Descent on the A2C (*Figure 3*). Afterwards, in order to encourage the network to do exploration, authors have developed the entropy penalty for exploration process. The discriminator updates are decoupled from the with usage of a replay buffer which serves as a communication layer between the actors and the discrimination network. With this implementation, authors have explained that the latter of the network is optimized at a higher rate than training of the policy network due to the sizes of the networks.

In each of the forward passes in the network, the discriminator takes in either a fake image pair or a real image pair so that agents will be encourage to make more similar renderings to the given target image to be able to fool the discriminator network which thereby enhances the agent to draw images more realistically as well as optimally.

# Evaluation and Results

The authors have tried their generative agents’ framework on the ImageNet dataset to observer whether the agents work on images that are not cropped or aligned well. They had downscaled 1000 images to 64x64 for training each agent. They have also trained the agents based on the classes of the images. As in the *Figure 4* the agents were able to reproduce the rough colors

A vase of flowers on a table

Description automatically generated

*Figure 4: ImageNet Evaluation of Agents*

and sometimes be able to extract the high-level features like the radial lines in the daisies.

The authors also evaluated on the Omniglot dataset. They have trained the generative agents to reconstruct the characters. Therefore, the agents have learned the mappings through bitmap representations of the strokes. As in the *Figure 5* the results were mostly successful in the reconstruction of the characters. Authors have indicated that they have introduced a number of hyper-parameters just for this data set in order to encourage the agent to learn more natural behaviors. More specifically, agent received rewards from the environment by a small penalty for each time a stroke is been made as well as the time it takes for that stroke. Without these penalties, they have indicated the agents achieve almost perfect results yet with un-natural movements.

Although authors evaluated their agents in the mentioned two datasets, the main dataset that they trained and evaluated their agents is called CelebA-HQ. Agents were never received the human drawings of the images but only the realistic photographs of human faces. They have divided their results into two portions as the short episodes and long episodes because agents that are trained with the short episodes learned qualitatively different policies than the ones in the long episodes since agents didn’t had numerous interactions. *Figure 6* shows how these agents trained on short episodes performed. Those agents were able to do a simple abstraction of the faces in the images. Short episodes took 17 steps in varying brushes and action spaces.

A close up of a keyboard

Description automatically generated

*Figure 5: Omniglot Evaluation of Agents (Left are the original characters, Right are the agents’ reconstructions)*

# A picture containing photo Description automatically generated

*Figure 6: Results of Agents on Short Episodes*

The *Figure 7* shows the agents that are trained on longer episodes which were able to successfully extract most of the features and reconstructed the images optimally. As in all cases agent was consistent with the architecture, only variety of the characteristics in the environment created diversity in the styles of the reconstructions. Agents were able to manipulate the location, color and thickness of the brush through the strokes with a high accuracy. Due to the reinforcement learning’s objective of maximizing the delayed rewards, policy of the agents deviates because of the greediness in the policy and take actions that reduce the quality of the image in the early steps in the episode.

To be able to run the long episodes Temporal Credit Assignment holds a crucial role. Without TCA agents start to make meaningful actions in the last 15 to 60 steps of the episode’s despite from the episode length. However, agents were able to make use of the full episode up to 1000 steps with a different qualitative generation policy through the TCA usage. To obtain qualitative results authors have used a varying discount factor scale from 0.9 to 0.99 in those episodes. Through *Figure 8* and *Figure 9* results of the training curves are given.

A group of people posing for the camera

Description automatically generated

Figure 7: *Results of Agents on Long Episodes*

# A close up of a map Description automatically generated

*Figure 8: Left curve indicating mean discriminator reward obtained by generators. Right curve shows the Mean L2 distance to the final generated image to the target image*

# Discussion

As the authors have mentioned in their paper, they have tried to make their network work on every environment when the enough time is given. This is one of the biggest improvements from the original SPIRAL. Even though this is crucial, they haven’t indicated the time required to train the agents through both the new environments (Omniglot & ImageNet) they have tested as well as the original environment (CelebA-HQ). Therefore, we cannot be sure that the training of these agents takes an optimal amount of time. Although, this SPIRAL++ network can be scaled to be used in protein-folding, chemical synthesis etc. as the authors have indicated, it’s not clear that this would be efficient in these cases since they haven’t tried any scaling with this framework for other areas of expertise. In addition to that, for their new environment tests such as the ImageNet & Omniglot dataset they had changed the hyper-parameters according to these data. Therefore, this implies that their train isn’t generalized for every environment, for the new environments they require to make some fine-tuning. This makes question marks on the readers mind through the viability of the solution about whether the framework is generalized or not.

# Conclusion

In summary, authors were able to generalize the original SPIRAL with the modifications through changing the reward system by giving rewards to the agent at each step instead of at the end of the episodes. They also mention with this new SPIRAL++ implementation they were able to generalize their trained framework to be used in any of the environments. Even though the

*Figure 9: Left curve indicating L2 distance of the canvas to its target image for 5 different episodes with discount 0.99. Right curve shows amount of improvement in L2 difference through each*

results they achieved from both the new environments and the original CelebA-HQ dataset were highly successful, which the results are shown in the previous sections, they had done modifications to achieve some of these precise results in the new environments. I believe that these modifications is one of the biggest weaknesses of the approach in the paper since they change the discount factor from environment to environment as well as some other hyper-parameters they don’t mention. This conflicts with the generalization of the problem. Another hold-back of this paper is I believe the training and evaluation times of the framework. It might be taking optimal or sub-optimal time to train the framework as well as the evaluation, yet the authors haven’t mentioned anything about it, which I presume takes a lot of time to train and evaluate both the agents and the framework. Even the computation time is very long it should’ve have been indicated for the readers comprehension.

Despite the weaknesses, the agents were able to imitate the human-faces as well as other objects precisely in the artificial intelligence levels. I wasn’t expecting a reinforcement agent to be successful at interpreting with the images while extracting the features of an image. Additionally, the SPIRAL++’s new reward system highly improves the agent’s precision at drawing. I think that rewarding at each step instead of each episode is much more an optimal approach allowing the agent to perform better. As the results suggest agents were performing in a noteworthy accuracy

![A close up of text on a white background

Description automatically generated]()