International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

# Vector Sketch Generation using Variational Auto Encoders

Dhiraj Sharma S, Balaji B, Satya Sai Krishna
SRM institute of Science and Technology
Kattankulathur, Tamil Nadu

***Abstract*:- Variational auto encoders and auto-regressive decoders are used to construct stroke based drawings of common objects both unconditional and conditional.The model will be prepared on an informational index of human drawn pictures of various classes. This additionally causes the model to sum up ideas in a similar way as people do. A Varia- tional Auto encoder model will be utilized for accomplishing the equivalent. We utilize a vector [P1,P2,P3].If P1 exists then the pen is down (on the canvas), in the event that P2 is exists, at that point Pen is up, and on the off chance that P3 is exists, at that point that implies that sketch has finished. This model has the capability to complete existing but fragmented portrayals. There are three sorts of strokes recorded. By inserting between latent vectors, we can imagine how one picture transforms into another picture by envisioning the recreations of the interpolations. It has numerous potential applications, from helping the innovative procedure of a craftsman, to helping teach students how to draw.**

***Keywords: RNN, VAE, Vector Image Generation, Sequence-to-Sequence AE***

## 1 INTRODUCTION

Sketch is a pictorial portrayal of considerations, pictures or mental picture in our psyches. From a youthful age, we build up the capacity to convey what we see by means of illustrations on paper with the help of pencil or colored pens. Numerous endeavors have been made to generate a sketch utilizing Generative Adversarial Networks[6]. The issue with these models lies in the manner in which they handle sketches. The vast majority of these models handle sketches as a matrix of pixels. Therefore the model may most likely create arbitrary sketches, yet it will never comprehend sketching. Not at all like machines, which manage low goals pixel lattice, Humans build up an abstract concept of what drawing is, they don't see the sketches as certain pixels obscured on a white pixel matrix. For people, it's a few lines, bends, shapes,depth, shading ,which characterize what a sketch is. People don't investigate a sketch as some network and after that contrast it with different frameworks they've seen before to conclude that it's a sketch of a feline or a puppy. They can say what the sketch is about, just by taking a gander at it. This is on the grounds that they know how a canine or a feline is drawn, what number of lines, what number of circular segments it takes to draw one. The itemizing of face done by various strokes. These are a portion of the highlights that assist people in distinguishing a sketch. Along these lines the manner in which people decipher a sketch is not the same as a machine. In this work, we present a model which produces sketches, however comprehends sketching. This can be exhibited by the sketches it produces, they are not flawless as the model attempts to sketch on it's own. our goal is to train the model to draw and sum up abstract illustration concepts a similar way people do. The latent vector delivered by encoding utilizing VAE can create some intriguing outcomes when added with some other latent vector of the equivalent or distinctive class.

The Recurrent Neural Network(RNN) based generative model is fit for producing diverse kinds of representa- tions in vector group which corresponds to the pen strokes at each time step. The model utilized is like succession to sequence to sequence encoder introduced by google and furthermore is an improvement of sketchRNN by google[4] also known as neural representation of sketch diagrams.The dataset utilized by our model is additionally an openly accessible dataset called quickdraw dataset , developed by google. it can be utilized very well by using their official API.The preparing method is one of a kind as vector draws the sketch making it more vigorous than other preparing methodology which are typically generalized. Potential Future uses of this model are likewise talked about in this paper.

## 2 RELATED WORK

There is a long history of work related to algorithms that mimic painters.Drawing and Animation Using Skeletal Strokes [13] published in year 1994 introduced an algorithm which emphasizes on vector graphics realization of the brush and stroke metaphor using arbitrary pictures as 'ink'. localized parametric coordinate system transformation along the stroke application path was used by the author to train a model which can generate fine strokes for any sketch mimicking the exact behavior of some artist on which it was trained. This shows that the idea to generate sketches by machine has been in existence for a long time. This is because, it not only shows the capabilities of modern machines but also helps several artists with their creative thinking process. skeletal strokes can be used to create rich and complex drawings from scratch. This was a great feat considering that it was done back in 1994. I still inspires several people to build models which can generate sketches from scratch. However, it depends too much on source image as it has to be deformed to obtain a single stroke vector. This reduces the system's capability to draw with innovation. Not just that but it also lacks the capability to mimic several artists nor does it allows for observation of different sketches formed by interpolating the latent vectors.


(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

Generating Sequences With Recurrent Neural Networks [1] is another one of the bases that helps form our model. Published in 2014, it states that if a complex sequence can be generated with long-range structure, then it's extended to generate handwriting synthesis. The author uses Synthesis Network, RNN(Recurrent Neural Network) and LSTM(Long Short-term Memory) to synthesize diverse and realistic samples of online handwriting.One of the major disadvantage of this model is that the network sometimes has trouble determining the alignment between characters and the trace.

Stroke Based Stylization Learning and Rendering with Inverse Reinforcement Learning [10], published in 2015 uses Policy Gradients with policy based Exploration(PGPE) with Inverse Reinforcement Learning to learn an artist's style exclusively. So the trained model will be able to depict his style. However this model is highly over fitted to that artist(justified,given its purpose) but it prevents the model from generalizing concepts of drawing.Authors use Inverse Reinforcement learning to analyze the artist's style and reproduce sketches in the same style.One of the main advan- tages of this approach is that Sketches specific to certain artists can be obtained due to the use of PGPE based Inverse RL. However it also faces a serious drawback due to this as mentioned earlier.

Ladder Variational Autoencoders [2] by casper, uses variational autoencoder to construct deep generative model and produce a structured high-level latent representations of the given sketch.In it's core, it uses Multi Layer Perceptrons (MLP)mapping and Batch normalization to achieve this.It puts much more importance to higher layers which gives the better performance of the model and provides a tighter bound on the log-likelihood however, the Batch normalization gives the additional noise ,and is not fully understood and deserves further investigation.

Sampling Generative Networks[15] by Tom White introduces several techniques for sampling and visualizing the latent spaces of generative models. For deriving attribute vectors, bias-corrected vectors with data replication and syn- thetic vectors with data augmentation are also introduced in this paper. This paper lays most of the base work required for interpolation of latent vectors which can also be used in our model to understand even more deeply as to how well it understood the sketch.It takes dot product of various latent vectors with attribute vector to produce vectors which can be changed arithmetically.Major disadvantage of this model is that interpolations are not linear and it results in the increased complexity of operations performed on it.

Sketch-pix2seq: a Model to Generate Sketches of Multiple Categories [19] published in 2017, presents a sequence-to- sequence Variational auto-encoder (VAE) model called sketch-RNN which is able to generate sketches based on human inputs. The sketch-pix2seq model can learn and generate a multiple categories of sketches.It uses CNN as the encoder and captures the local structure of the images.Four models i.e, RNN+KL, RNN-KL, CNN+KL, CNN-KL performance are compared in this model used by the author. Since it is learning multiple categories simultaneously it will help save computational resources. And no extra constraint is put on the encoder to learn the posterior latent space.Models with KL-divergence tend to produce sketches of an unexpected third category during interpolation, which implies that mod- els with KL-divergence are unsuitable for learning multiple categories. This has to be taken into consideration while designing the model.

ShadowDraw:Real-time user guidance for free hand drawing [18] by Yong Jae lee is an enhancement of Sketch RNN proposed by People at google brain, and focuses primarily helping users draw by working in an interactive mode.It Provides guidance for freeform drawing. Based on user's drawing progress, ShadowDraw retrieves relevant data from a database, and provides suggestions. A hashing technique enforces both local and global similarity and provides suffi- cient speed for interactive feedback. the working of model is as follows: Through the user input, the system runs Edge extraction, to recognize the pattern of the sketch. Patch descriptors then determine the edge positions, and through Min- Hashing, relevant data is compared and suggested.The results from ShadowDraw produces realistically proportioned line drawings. The user Interface is given importance for better user experience.The system fails in case of people who have poor drawing skills as edge extraction becomes difficult, and relevant data cannot be retrieved from the database.

Learning to Doodle with Deep Q Networks and Demonstrated Strokes [16] by Tao Zhao is another of the models that tries to integrate deep learning to produce vector sketches. At first,the model learns to draw simple strokes by imi- tating in supervised fashion from a set of stroke action pairs and then try to draw more without GT(ground truths).The authors call it SDQ,(Stroke demonstration and Deep Q Learning).In this model, use of reinforcement learning allows the model to be trained far more and will be able to generate sketches that normally wouldn't be possible. However, It requires additional supervised learning in the beginning , which consumes more time and additional Strain on GPU.

Sketchsegnet: A RNN model for labeling Sketch Strokes[17] published in 2018,treats the problem of stroke-level sketch segmentation as a sequence-to-sequence generation problem, and a recurrent neural networks (RNN)- based model Sketch SegNet is presented to translate sequence of strokes into their semantic part labels. sketches with com- plex initial structures have have decreased stroke based accuracy in this model and is rectified in our proposed model. Another paper published in the same year,


(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

Unsupervised Image to Sequence Translation with Canvas-Drawer Networks [9] Generates images directly in a high- level domain (e.g. brush strokes), without the need for real pairwise data. It uses a Canvas- Drawer system. Canvas recreates x(n+1) given x(n) and y(n). Drawer minimizes the pixel wise distance between the final state of the renderer and target image.Optimizing for recreation through an interpretable constraint is a promising avenue for many unsuper- vised methods and needs to be refined more.The sequential drawer can only handle fixed length sequences. Also, this model heavily relies on pixel data and not vector data. These are some of the work which forms basis for our model. Our model Takes into account, all the advantages and disadvantages of these models and tries to balance them out. There are some qualities of certain models which cannot be reproduced in our model as they are specifically designed to fulfill some purpose. As mentioned earlier, our model's focus is to generalize concepts the same way humans do and produce sketches both conditionally and unconditionally.

## 3 VECTOR SKETCH GENERATOR

### *3.1 Architecture*

Our model is a Sequence-to-Sequence Variational Autoencoder (VAE), similar to the architecture described in [3, 12]. Our encoder is a bidirectional RNN [8] that takes in a sketch as an input, and outputs a latent vector of size $N_z$. Specifically, we feed the sketch sequence, S, and also the same sketch sequence in reverse order, $S_{reverse}$, into two encoding RNNs that make up the bidirectional RNN, to obtain two final hidden states:

h = encode (S), h = encode ($S_{reverse}$), h = [ h ; h ]

The last concatenated state got is taken and anticipated into two vectors $\mu$and $\sigma$, each of size $N_z$,using fully connected network.A random vector Z is constructed by using $\mu$, $\hat{\sigma}$ and N(0,I),a vector of Gaussian IID variables of size $N_z$.

$$\mu = W_\mu + b_\mu$$
$$\hat{\sigma} = W_\sigma h + b_\sigma,\ \sigma = \exp(\hat{\sigma})$$
$$z = \mu + \sigma \odot N(o,I)$$

Under this encoding scheme, the latent vector z is not a deterministic yield for a given input sketch, but for a random vector.
Our decoder is an autoregressive RNN that outputs a sketch from given sample which is conditional to latent vector
z. we apply tanh operation to ensure that standard deviation values are non-negative.One of the significant challenges that our model faces is to understand when to stop sketching. Since the probabilities of occurrence of each pen stroke (up,down,halt) is highly unbalanced, it becomes more diligently to train.How we conquer this is mentioned in our upcoming training section. Subsequent to training, we can start sampling sketches from our model.during this process, we generate the parameters for both GMM and categorical distributions at each time step and sample a result $S'_i$ for that time step.We keep the sampling process going on till p3 =1(halt stoke) or when i = $N_{max}$(largest sketch sequence).

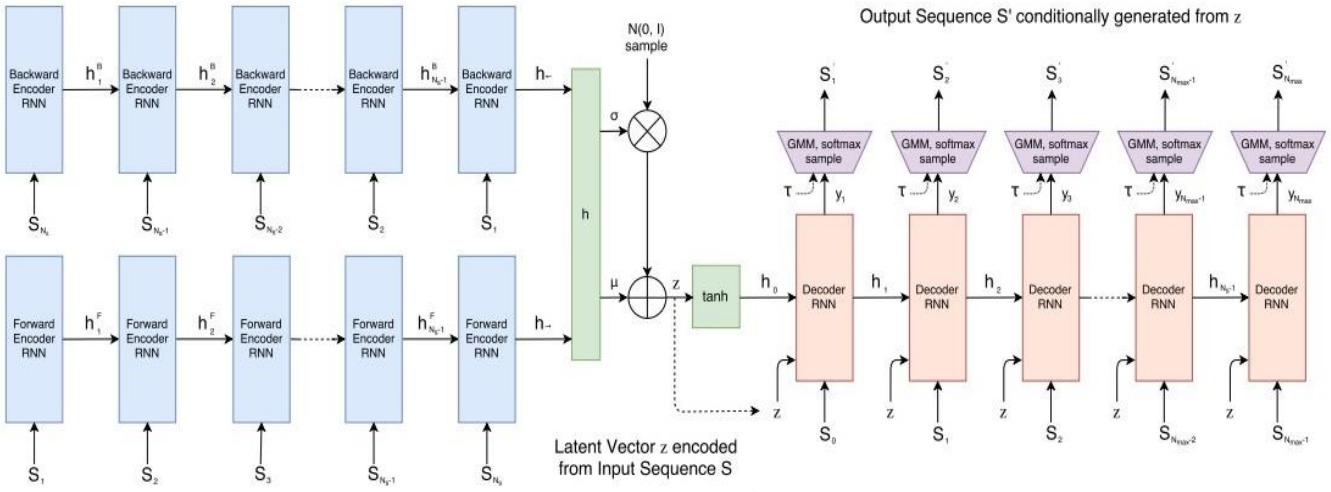


Figure 1: Schematic diagram of the model

### *3.2 Training*

Our preparation system pursues the methodology of the Variational Autoencoder [3], where the loss work is the aggregate of two separate losses: the Reconstruction Loss and the kullback-Leibler Divergence Loss. Our model is prepared by limiting these two loss capacities. Reconstruction loss is the distinction between the example sketch vector created by the decoder and the first info vector.Note that we dispose of the pdf parameters demonstrating the (Wx,Wy) points beyond $N_S$ when calculating $L_S$, while $L_P$ is calculated using all of the pdf parameters modelling the (p1, p2, p3) points until $N_{MAX}$. Both terms are normalized by the total sequence length $N_{MAX}$.We discovered this system of loss figuring to be increasingly hearty and enables the model to effortlessly realize when it should quit drawing, not at all like the prior referenced technique for doling out significance weightings to p1, p2, and p3.


(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

The Kullback-Leibler divergence loss in this model is determined by finding the contrast between the appropriation of inactive vector z and that of IID Gaussian factors' vector with zero mean and unit variance.

$$L_{KL} = -\frac{1}{2N_z} 1 \quad \hat{\sigma} \quad \mu^2 \quad exp\ \hat{\sigma}$$

Overall loss is the weighted sum of both these loss functions
$Loss = L_R + W_{KL}L_{KL}$

As $W_{KL} \to 0$, our model approaches pure autoencoder sacrificing its ability to enforce a prior over our latent space while obtaining better reconstruction loss metrics.As such, it's a trade off between minimizing one loss over the other.

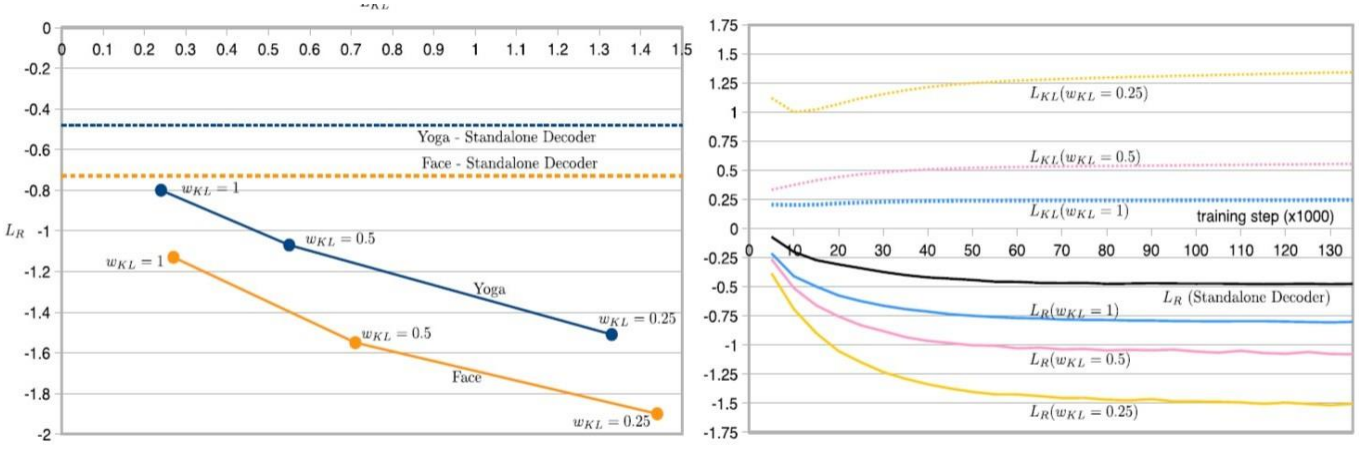


Figure 2: Tradeoff between $L_R$ and $L_{KL}$, for two models trained on single class datasets (left).Validation Loss Graph for models trained on the Yoga dataset using various $w_{KL}$. (right)

## 4 RESULTS

### *4.1 Conditional Reconstruction*

In Figure 4 (left), we think about a few Reconstructions for feline class. From the figure it very well may be seen that outlines have comparative properties as the information picture, and infrequently include or expel subtleties, for example, a hair, a mouth, a nose, or the introduction of the tail.We likewise demonstrate a model prepared on the pig class, as appeared in Figure 4 (right).

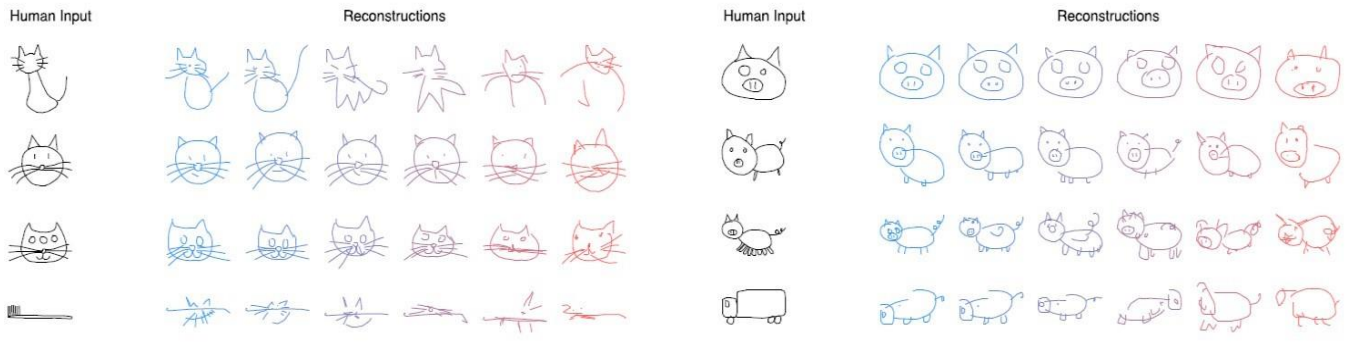


Figure 3: Schematic diagram of the model

### *4.2 Interpolation*

By adding two latent vectors delivered by input sketches from two unique classes, one can see how one transforms into the other. In the given figure, we interject a feline and pig sketch with different $W_{KL}$ settings. Subsequent to rehashing the examination for a few hyperparameters,it was seen that higher $W_{KL}$ delivered progressively more coherent interpolated sketches.

A model prepared on higher quality representations may discover its way into instructive applications that can help show


(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

understudies how to draw. Indeed, even with the straightforward portrays in QuickDraw, we have turned out to be significantly more capable at illustration creatures, creepy crawlies, and different ocean animals after leading these analyses. A related application is to encode an unrefined, ineffectively outlined illustration what's more, produce all the more stylishly looking propagations by utilizing a model prepared with a high wKL setting to deliver a progressively sound form of the illustration. In the future, we can likewise examine expanding the latent vector towards the path that augments the style of the illustration by fusing client rating information into the preparation procedure. Consolidating hybrid variations of sequence-generation models with unsupervised, cross-domain pixel image generation models, for example, Image-to-Image models [5, 9, 14], is another energizing bearing that we can investigate. We would already be able to consolidate this model with directed, cross-space models such as Pix2Pix [11], to infrequently produce photograph practical feline pictures from created representations of felines. The other way of changing over a photo of a feline into an unreasonable, however comparative looking sketch of a feline made out of a negligible number of lines is by all accounts an all the more fascinating issue.

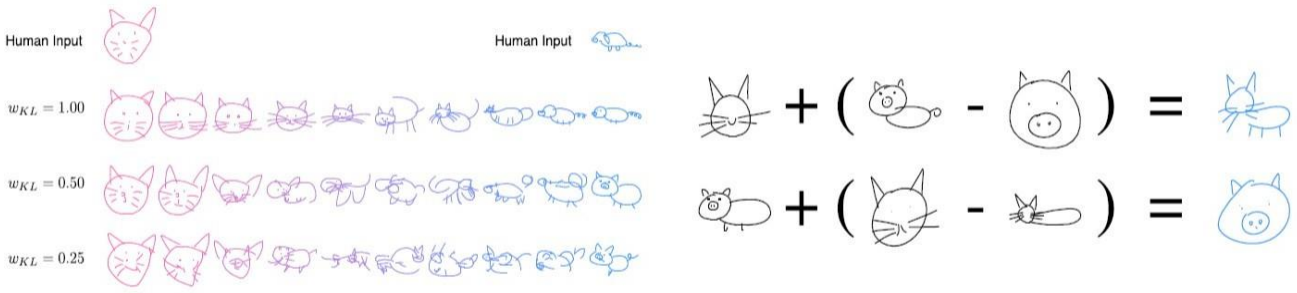


Figure 4: Latent space interpolation between cat and pig using with various $W_{kl}$ settings.

## 5 CONCLUSION

In this work, we build up a philosophy to demonstrate sketch illustrations utilizing recurrent neural networks.Our model can complete existing inadequate portrayals. Our model can likewise encode existing portrayals into a latent vector, and create comparable looking representations adapted on the latent space.We additionally tell the best way latent vectors can be interpolated and what we can anticipate from the cognizant representations created accordingly. The dataset which we are at present utilizing is made and kept up by google, and expansion to their dataset will expand the productivity of the model. Since it's a freely accessible dataset, Anyone can utilize their dataset and accomplish something imaginative utilizing that in the territory of generative vector image modelling.

## 6 REFERENCES

[1] A. Graves. Generating sequences with recurrent neural networks. arXiv:1308.0850, 2013.
[2] C. Kaae Sønderby, T. Raiko, L. Maaløe, S. Kaae Sønderby, and O. Winther. Ladder Variational Autoen-coders. ArXiv e-prints, Feb. 2016.
[3] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. ArXiv e-prints, Dec. 2013.
[4] David Ha, Douglas Eck. A Neural Representation of Sketch Drawings. ArXiv:1704.03477v4, May. 2017.
[5] H. Dong, P. Neekhara, C. Wu, and Y. Guo. Unsupervised Image-to-Image Translation with Generative Adversar- ial Networks. ArXiv e-prints, Jan. 2017.
[6] I. Goodfellow. NIPS 2016 Tutorial: Generative Adversarial Networks. ArXiv e-prints, Dec. 2017.
[7] Kevin Frans, Chin-Yi Cheng. Unsupervised Image to Sequence Translation with Canvas-Drawer Networks. arXiv cs.CV, 2018.
[8] M. Schuster, K. K. Paliwal, and A. General. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing, 1997.
[9] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised Image-to-Image Translation Networks. ArXiv e-prints, Mar. 2017.
[10] Ning Xie, Tingting Zhao, Feng Tian, Xiaohua Zhang, Masashi Sugiyama. Stroke Based Stylization Learning and Rendering with Inverse Reinforcement Learning. IJCAI 2015.
[11] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. ArXiv e-prints, Nov. 2016.
[12] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Józefowicz, and S. Bengio. Generating Sentences from a Continuous Space. CoRR, abs/1511.06349, 2015.
[13] Siu Chi HSU, Irene H. H. LEE. Drawing and Animation Using Skeletal Strokes. SIGGRAPH 1994.
[14] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim. Learning to Discover Cross-Domain Relations with Generative Adversarial Networks. ArXiv e-prints, Mar. 2017.
[15] T. White. Sampling Generative Networks. ArXiv e-prints, Sept. 2016.
[16] Tao Zhou, Chen Fang, Zhaowen Wang, Jimei Wang, Byungmoon Kim, Zhili Chen, Jonathan Brandt, Demetri Terzopoulos. Learning to Doodle with Deep Q Networks and Demonstrated Strokes. arxiv cs.cv, 2018.
[17] Xingyuan Wu,Yonggang Qi, Jun Liu, Jie Yang. Sketchsegnet: A RNN model for labeling Sketch Strokes. IEEE International Workshop on Machine Learning for Signal Processing, 2018.
[18] Y. J. Lee, C. L. Zitnick, and M. F. Cohen. Shadowdraw: Real-time user guidance for freehand drawing. In ACM SIGGRAPH 2011 Papers, SIGGRAPH '11, pages 27:1–27:10, New York, NY, USA, 2011. ACM.
[19] Yajing Chen, Shikui Tu, Yuqi Yi, Lei Xu. Sketch-pix2seq: a Model to Generate Sketches of MultipleCategories. arXiv cs.CV, 2017.


(This work is licensed under a Creative Commons Attribution 4.0 International License.)