Design a Loss Function which Generates a Spatial configuration of Image In-betweening

Paulino Crsitovao[*1] Hidemoto Nakada[*2*1] Yusuke Tanimura[*2*1] Hideki Asoh[*2]

[*1] University of Tsukuba

[*2] National Institute of Advanced Industrial Science and Technology of Japan

Instead of generating image inbetween directly from adjacent frames, we propose a method based on inbetweening in latent space. We design a simple loss function which generates a latent space that represent the spatial configuration of image inbetween. Contrary to the frame based methods, this model can make plausible assumption about the moving objects in the image and can capture what is not seen in the images. Our model has three networks, all based on variational autoencoder, sharing same weights. We validate this model on different synthetic datasets. We show the details of our network architecture and the evaluation results.

1. Introduction

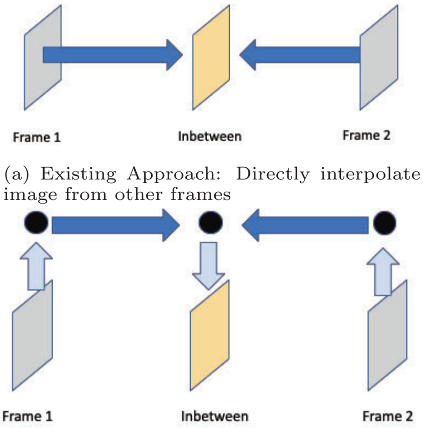
For machines to become more intelligent and autonomous is essential that they understand the world around them, by being able to learn and understand the semantics present in the data. One way to approach this issue is by using generative models. These models can learn the patterns present in the data and generate new similar sample. This work seeks to discover latent representations present in data also design an objective function which generates the spatial configuration of image inbetween. Image inbetween attempts to generate image interpolation from nearby frames. The generated image has to preserve the spatial configuration of the moving objects. Up to now, optical flow [Yi 15],[Mémin 98] and convolutional neural networks [Amersfoort 17] have been proposed to generate image interpolation. Both methods generate image inbetween directly from adjacent frames 1. The result is blur images and loss of contextual information, also they cannot capture what is not present in the the frames. When generating image inbetween preserving the spatial location, shape, color is relevant for some application, for this reason we design a simple model that is able to preserve the contextual representations of objects between nearby frames. This model find scope in several areas such as movie and animation industries where they have to draw each individual frame and in image inpaiting.

In section 2 we describe our proposed model to generate image interpolation, in section 3, we show the results and we present conclusion in the last section.

2. Proposed Method to Generate Image Inbetween

2.1 Model Overview

Our model is based on generative models, which have shown tremendous success in different field such as pattern recognition, image classification, natural language process and reinforcement learning. The proposed approach uses variational autoencoder to generate image inbetween.



(a) Existing Approach: Directly interpolate image from other frames

(b) Latent Model Interpolation

Figure 1: Comparison between existing approach and our

2.2 Proposed Loss Function to Generate Image Inbetween

Evaluating generative models means adjusting the internal weights of the network in order. to minimize an error measure. The error is usually given by a loss function. We optimize our network to minimize the distance between the image inbetween and ground truth in latent space. We follow a standard loss function for variational autoencoder. The model has three VAE each with its error function, we sum all errors function plus a an error function caused by the difference between the average latent space of the nearby frames and ground truth. We introduce a scalar hyper-parameter that we call coefficient α (below equation). The coefficient α is an adjustable parameter which express how much is relevant the difference between the Z1 and Z'. Next section we highlight the relevance of α.

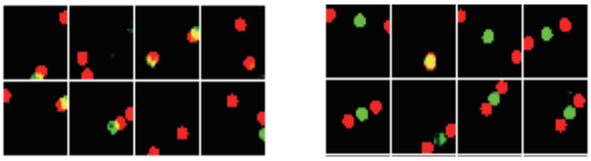
Contact: paulinocristovao86@gmail.com

$$\mathbf{l}_{(\mathbf{X_0},\mathbf{X_1},\mathbf{X_2})} = \mathbf{l}_{\mathbf{VAE}}(\mathbf{X_0}) + \mathbf{l}_{\mathbf{VAE}}(\mathbf{X_1}) + \mathbf{l}_{\mathbf{VAE}}(\mathbf{X_2}) + \alpha(\mathbf{D}_{\mathbf{KL}}(\mathbf{q}_{(\mathbf{X_1})}||\frac{\mathbf{q}_{(\mathbf{X_0})} + \mathbf{q}_{(\mathbf{X_2})}}{\mathbf{2}}))$$

2.3 Effects of Coefficient α

The adjustable hyper-parameter α modifies the traditional variational autoencoder objective function. It places a restriction on the latent space. This coefficient constraint the latent representations to generate a latent space which represent the spatial configuration of inbetween objects in the image. For $\alpha = 0$ represents the traditional VAE, no restriction is placed in the latent model, increasing the value of α means increasing restrictions on the latent representations.

When evaluating images, the motion of large objects seems easy to evaluate however, evaluating small motion is more complex. We aim to be able to detect small and large changes between frames.



(a) $\alpha = 0$,epochs=5000 (b) $\alpha = 100$,epochs=5000

Figure 2: Inbetween Image: Red dots: Nearby Images; Green dot: Inbetween Image

3. Experiments

3.1 Data Preprocessing

We used synthetic datasets. Two scenarios were tested: first where the object has one variable influencing its rotation which we name "one degree of freedom" and second having two variables "two degrees of freedom". The images were reshape into size of 32x32. For training and testing we randomly sample a triplet images by giving a certain interval among the frames.

3.2 Network Implementation

The base of our model follows a variational autoencoders (VAE), the network model has three VAEs 3, all sharing same weights to reduce the number of hyper-paremeters. The encoder has four convolutional layers, first layer (128 nodes), second(256 nodes), third (512 nodes), fourth (1024 nodes), kernel size = 4 and stride 2. The decoder has four deconvolutional layers, first layer (512 nodes), second (512 nodes), third (256 nodes), fourth (64 nodes) with same kernel size and stride. We input a triplet image. For this work we ignore the output of the nearby frames 3.

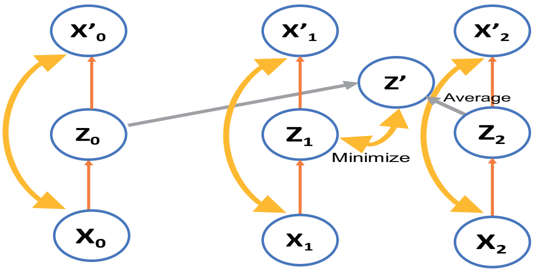


Figure 3: Network implementation

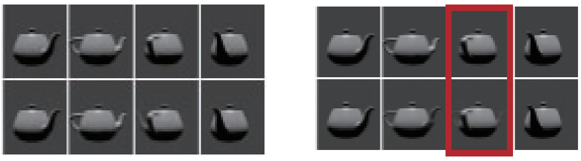
3.3 Reconstruction

The goal is to test the location accuracy

In this section, firstly we qualitatively demonstrate that our proposed model can reconstruct the input image. We tested the reconstruction object location, shape and color. Two scenarios is tested, on α equal to zero and α greater than zero.

3.3.1 One degree of freedom

Below results are for testing. We note that after strong coefficient $\alpha = 100$, the reconstruction test misses some features of the input data.

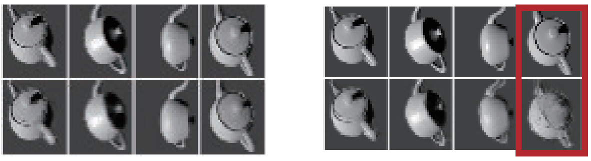


(a) $\alpha = 0$, epochs=10000 (b) $\alpha = 100$,epochs=10000

Figure 4: Teapot:1st row: Original Image, 2nd row:Reconstructed Image. Red box shows imperfect object reconstruction

3.3.2 Two degrees of freedom

We increase the complexity of the data, the rotation of the object is influenced by two variables.

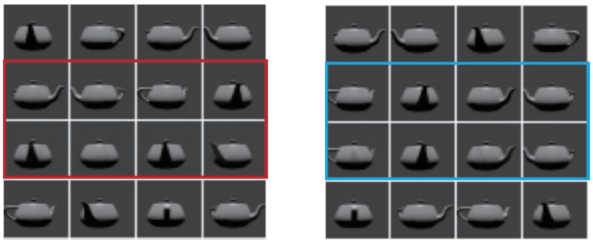


(a) $\alpha = 0$, epochs=10000 (b) $\alpha = 100$,epochs=10000

Figure 5: Teapot: 1st row:Original Image, 2nd row:Reconstructed Image

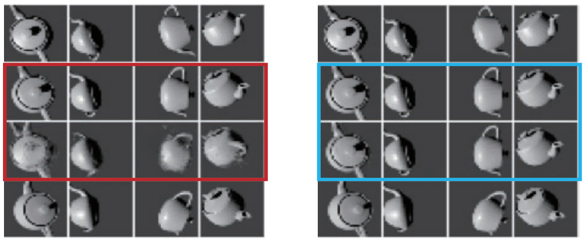
3.3.3 Multiple Objects

The task of reconstructing 3 objects seems complex for the model, since it has to capture the pattern of each object and make correspondent matching while interpolating.



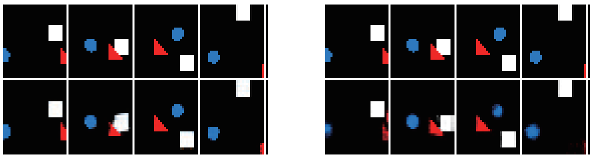
(a) $\alpha = 0$,epochs=10000 (b) $\alpha = 100$,epochs=10000

Figure 7: Teapot-Testing: 1st row:first image, 2nd row: ground truth, 3rd row: Inbetween Image, 4th row: second image. The red square box shows that with $\alpha = 0$ we have imperfect inbetween, Blue box show the correct inbetween



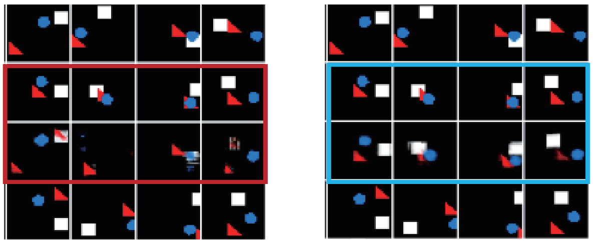
(a) $\alpha = 0$,epochs=15000 (b) $\alpha = 100$,epochs=15000

Figure 8: 2D Teapot-Testing: 1st row:first image, 2nd row: ground truth, 3rd row: Inbetween Image, 4th row: second image. The red square box shows that with $\alpha = 0$ we have imperfect inbetween, Blue box show the correct inbetween



(a) $\alpha = 0$, epochs=15000 (b) $\alpha = 100$,epochs=10000

Figure 6: Mulitple Objects: 1st row:Original Image, 2nd row:Reconstructed Image



(a) $\alpha = 0$,epochs=15000 (b) $\alpha = 5$,epochs=15000

Figure 9: Multiple Objects - Testing: 1st row:first image, 2nd row: ground truth, 3rd row: Inbetween Image, 4th row: second image

3.4 Image Inbetween

As the model was able to reconstruct its input image, even with strong restriction placed in the latent space, next we qualitatively demonstrate the image inbetween generated by our model 7. Using two nearby images with large displacement from one image to other, with zero coefficient the image inbetween does not preserve the accurate spatial location of the object, increasing the coefficient the model is able to generate the perfect image inbetween as we will show in the next section.

3.4.1 One degree of freedom

We trained the framework with images or rotating on x-axis with one degree of freedom. The testing size is 360, the test images used in training and testing are distinct. The images generated by our approach $\alpha = 100$ presents a fair inbetween 7.

3.4.2 Two and Six degrees of freedom

Previous examples we rotated the object in 360 degrees on x-axis, i.e. with one degree of freedom. It is easy to find the pattern of the data points as there are just 360 options or angles. We increased the complexity of the images by moving the object with two and six degrees of freedom. The results for two degrees are credible 8, while for six degrees (3 objects), the model does not perform well on testing phase 9.

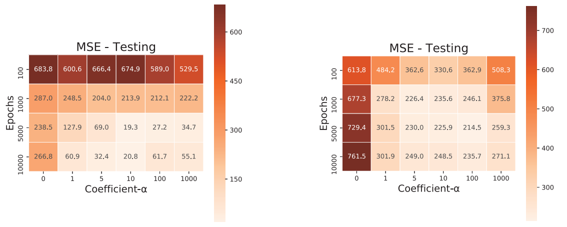
3.5 Quantitative Evaluation

The goal here is to evaluate the complexity of the dataset in terms of its degree of freedom. We evaluate the same object in one degree and two degrees of freedom. The results indicates that the two degrees of freedom is more complex. Its MSE gives higher values 10.

3.6 Linear Latent Space Interpolation

We sample pair of images x1 and x2 and project them into latent space z1 and z2 by sampling from the encoder, then linearly interpolate between Z1 and Z2 and pass the intermediary points through the decoder to plot the input-space interpolations. The objective is to estimate the continuity in the latent space. Below figures show the generated smooth interpolation of two nearby points. The latent codes used to generate the nine intermediate images are equivalent to (P=0.9, to 0.1): We observe smooth transitions between pairs of examples, and intermediary images remain credible 11. This is an indicator that this model is not just restricting its probability mass exclusively around training examples, but rather has learned latent features that generalize well.

Linear latent space interpolation, indicate that there is a continuity in the latent space which allows a smooth interpolation. We show an example of 3 objects moving in random direction 12, we linearly interpolate the latent space and generate the possible trajectory between first frame and last frame. This model can predict a long-term frames and has the ability to capture their trajectory.



(a) one degree (b) two degrees

Figure 10: MSE loss

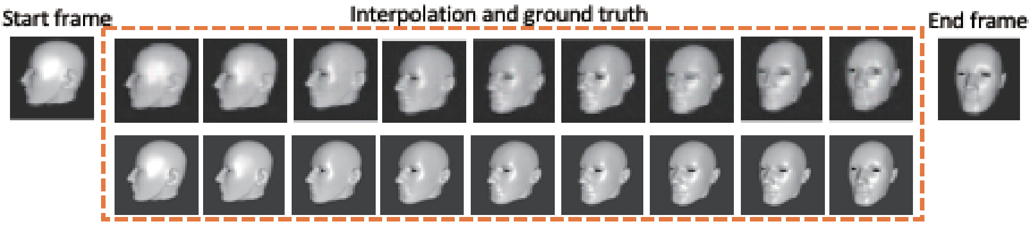


Figure 11: Linear Long-term Interpolation: Face turning to left side

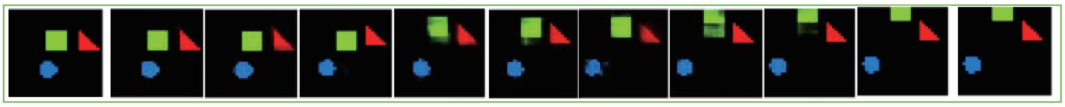


Figure 12: Linear Long-term interpolation of 3 objects, we see the smooth interpolation from first frame and the last frame.

4. Related Work

The intention of unsupervised methods is to uncover the underlying latent representation of the data. Recent works on VAE [Higgins 17][Chen 18] [Berthelot 18] focus in disentangled the latent representations. This approach finds great application in scenario where there is a need to distinguish different characteristics present in the data, for instance, skin color, head pose, facial expression. A disentangled representation can be useful for natural tasks that require knowledge of the salient attributes of the data, which include tasks such as face and object recognition.
Our prosed model does not disentangle latent representations, it simply learn the pattern present in the data.

4.1 Improving Interpolation

While generating interpolation two fundamental characteristics have to be preserved: intermediate points along the interpolation are indistinguishable from real one and provide semantic and smooth morphing [Berthelot 18] The late characteristic is hard to achieve, for that reason [Berthelot 18] purpose a model based in variational autoencoder which introduce a regularizer which encourages interpolated data points to appear more indistinguishable from reconstructions of real data points. It is important to make a clear distinguish between image interpolation generated by latent model and our model. These latent model approaches cannot be used for our target application for the following reason, the dataset used by these models present some variation of the data, for instance in case of celebrity dataset mentioned earlier, it has many factors such as rotation of the head, skin color, age, gender, with or without glasses. The dataset we used does not present such characteristics in addition, we do not disentangle any specific factor of variations, we simply put a restriction on the latent model to generate an accurate image inbetween.

5. Conclusion

We present an alternative approach for generating an image inbetween by giving nearby frames which are non-consecutive images using a latent model. Our approach changes the Naive VAE objective function by introducing a hyper-parameter which constraint the latent representations. This model excels at predicting the image inbetwee in addition the model generalizes well for different datasets. For future, we will test this model on more complex data such as: Complex physical models, such as linked arms. Non-image data, for instance: text and audio data video i.e. video with fast motions and more moving objects. Finding better hyper-parameters between reconstruction and image inbetween.

Acknowledgement

This paper is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO). This work was supported by JSPS KAKENHI Grant Number JP16K00116.

References

[Amersfoort 17] Amersfoort, V., et al.: Frame Interpolation with Multi-Scale Deep Loss Functions and Generative Adversarial Networks, *arXiv preprint arXiv:1711.06045* (2017)

[Berthelot 18] Berthelot, D., et al.: Understanding and Improving Interpolation in Autoencoders via an Adversarial Regularizer, *arXiv preprint arXiv:1807.07543* (2018)

[Chen 18] Chen, T. Q., et al.: Isolating Sources of Disentanglement in Variational Autoencoders, *arXiv preprint arXiv:1802.04942* (2018)

[Higgins 17] Higgins, I., et al.: beta-vae: Learning basic visual concepts with a constrained variational framework, in *International Conference on Learning Representations* (2017)

[Mémin 98] Mémin, E. and Pérez, P.: Dense estimation and object-based segmentation of the optical flow with robust techniques, *IEEE Transactions on Image Processing*, Vol. 7, No. 5, pp. 703–719 (1998)

[Yi 15] Yi, C., Liyun, C., and Chunguang, L.: Moving Target Tracking Algorithm Based on Improved Optical Flow Technology, *Open Automation and Control Systems Journal*, Vol. 7, pp. 1387–1392 (2015)