Unsupervised Learning of Physical Variables from Video with Disentangling Autoencoders

Felix Meissen, felix.meissen@tum.de,
Felipe Peter, felipe.peter@tum.de

1. Introduction

The goal of this work is to train a neural network to learn physical properties of an object from video in an unsupervised manner, i.e., without using the ground truth of these properties. More specifically, we want the network to learn the horizontal (x) and vertical (y) position, velocity, and acceleration of a ball moving in front of a static background. The learned properties should be interpretable and also editable to generate new samples by manipulating the learned variables. For this task we use a β-Variational Autoencoder to obtain the underlying physical variables as latent variables in the bottleneck.

2. Related Work

Higgins *et al*. [4] proposed β-Variational Autoencoders (β-VAE) as a method of disentanglement to find the independent generative factors of a data set. Burgess *et al*. [1] successfully applied this method to the popular dSprites data set by Matthey *et al*. [8]. Iten *et al*. [5] showed that a β-VAE can be applied to recover different physical properties from time series of measurements of a physical experiment. While the works mentioned above perform disentanglement on still images or non-image data, to our knowledge there has not been any work on using disentanglement on video data.

3. Data Set

We create a synthetic data set, which contains sequences of images of a white ball with radius $r = 2$ pixels moving over a black background on a screen of size 64×64 pixels. The initial positions, velocities, and accelerations for all sequences are sampled from a uniform distribution to ensure that the generative factors of the data set are independent of each other. To ensure a significant impact of the velocity and acceleration on the trajectory, we do not sample values around 0. Figure 1 shows the trajectories and the correlation between the generative factors of an example data set with 500 sequences.

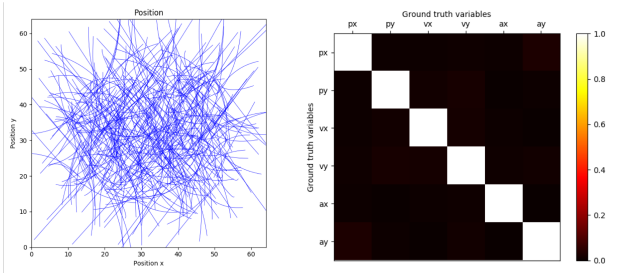

Figure 1. Left: Trajectories of a data set with 500 sequences. Right: Correlation between each pair of generative factors.

4. Methodology

4.1. Physical Question Answering

Question answering is an approach proposed in [5] to limit the knowledge of the encoder. The idea is visualized in Figure 2. Only the first n frames of a sequence are fed to the encoder to extract the relevant information. The question indicates the frame that has to be predicted and can be any frame of the full sequence. It is only passed to the decoder (see Figure 3 bottom for the network architecture), the encoder therefore has to provide a latent representation that contains all the information necessary (i.e., position, velocity, acceleration) for the decoder to answer the question.

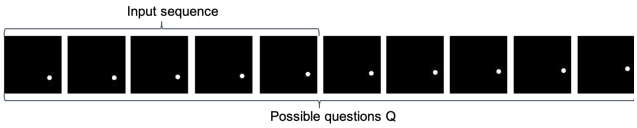

Figure 2. In this example, the sequence is 10 frames long. The first 5 frames are concatenated and fed to the encoder. The question can be any frame index of the whole sequence.

4.2. Physics Layer

We introduce a custom "physics layer" in the bottleneck of our autoencoder architecture (see. Figure 3 top). This layer computes the equation of motion to output the position in x and y at time step t, given as input the initial position, velocity, and acceleration in x and y (see Equation 1).

$$p(t) = p_0 + v_0 \cdot t + 0.5 \cdot a_0 \cdot t^2 \quad (1)$$

The intuition behind this approach is that by including this function directly in the bottleneck, we encourage the network to learn a representation of the latent variables which corresponds to the variables of the equation of motion. An end-to-end training is still possible because the equation of motion is differentiable. The physics layer can be combined with the question answering approach by using the question as a time step t for the equation of motion.

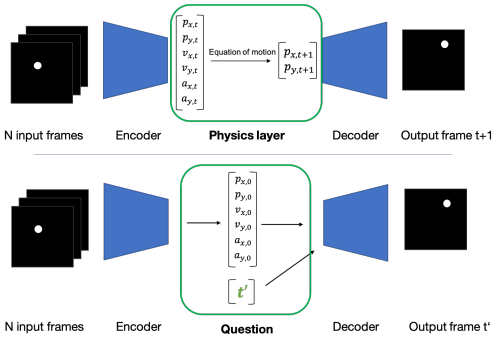

Figure 3. Schematic of an autoencoder with physics layer (top) and question (bottom).

4.3. Disentanglement

To obtain interpretable results, we choose the β-Variational Autoencoder from [4] as a method of disentanglement. This method enforces disentanglement by penalizing the KL-divergence between the distribution of the latent variables and a prior (in our case a Gaussian with zero mean and unit variance).

5. Experiments

We conduct experiments with the ideas from section 4. For this, we generate a data set as introduced in section 3 with 8192 sequences for training and 512 sequences for evaluation. Every sequence contains 15 frames of which 5 are used as input to the model.

Except for the bottleneck, the architecture we use is the same for every model. It follows an encoder-decoder architecture with one initial 2D convolution, three "skip convolutions" and two linear layers with ReLU activations in the encoder. The decoder mirrors the encoder. Our skip convolutions are inspired by the residual blocks from [3] and help the gradient flow through the network. Each model has a latent dimension of 6 which matches the number of generative factors in the data set. We use the Adam optimizer [6] with a learning rate of 5e-4 to train the networks for 600 epochs. The β-penalty was set to 1 for all experiments.

As a baseline, we trained a model with a question added to the bottleneck as in [5]. Next, we trained a model using our proposed physics layer in the bottleneck. This model only predicts the next frame $t+1$ after the input sequence. Finally we combined the two approaches in a third model which takes the question as an input to the equation of motion 1. We will refer to these three models as "Question-VAE", "Physics-VAE" and "Combined VAE" respectively from now on.

It should be noted that training all three models on the same data set is not a fair comparison, since the Physics-VAE only has to predict one frame in the future, the other two models up to 14 and thus have to estimate the velocity and acceleration way more accurately.

Our code and the trained models can be found at https://github.com/Mr-Pepe/dl4cv.

6. Evaluation

In this section we evaluate the qualitative and quantitative results of the training runs from section 5. For the evaluation, we create a second data set with 2000 sequences and the same properties as the training data set.

6.1. Quantitative Results

Correlation with ground truth

Figure 4 shows the correlation of the latent variables of the trained models with the ground truth generative factors of the evaluation data set. While it is hard to read any useful correlations from the Question-VAE, the correlations in the other two models are clearer.

The Physics-VAE seems to have learned only the position and mostly in x direction. Looking at Figure 5 we can see that the variances for the velocities and accelerations of this model are very high. This means that these variables do not hold a lot of information. We assume that the model already encodes the position at $t+1$ in the encoder and uses the identity of the position in the equation of motion to pass that information to the decoder.

The Combined VAE has successfully encoded the position and velocity in its latent space. This can be seen by looking at the variables 0 to 3 which have a strong correlation with px, py, vx and vy respectively and with no other ground truth factor.

Lastly, it seems that all the models failed to fully learn the acceleration as there is no strong correlation between any of the latent variables and the ground truth acceleration.

Intercorrelation

Figure 6 shows the intercorrelation between the latent variables of the trained models. The Question-VAE learns uncorrelated latent variables. Using only the physics layer leads to heavy intercorrelation in the latent space. The latents 0 to 3 of the combined model, which successfully

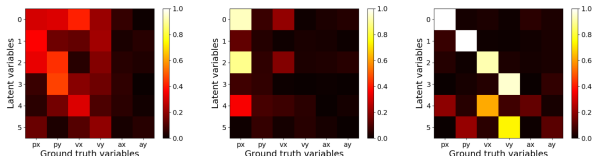

Figure 4. Correlation of the latent variables of the trained models with the ground truth generative factors of the data set. Left: Question-VAE. Middle: Physics-VAE. Right: Combined VAE.

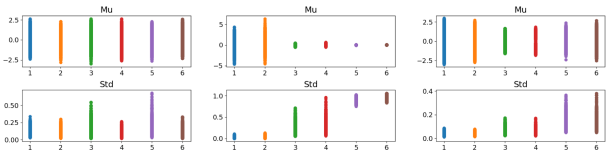

Figure 5. Distribution of the means and standard deviations of the latent variables for a subset of 2000 sequences. Left: Question-VAE. Middle: Physics-VAE. Right: Combined VAE.

learned the ground truth factors, have no intercorrelation, while the two other latents which don't correspond to any meaningful factor are strongly correlated.

Correlation however is not a sufficient metric to measure disentanglement because it does not inform about the interpretability of the latent variables.

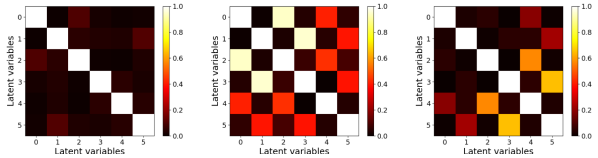

Figure 6. Inter-correlation of the latent variables of the trained models. Left: Question-VAE. Middle: Physics-VAE. Right: Combined VAE.

Disentanglement metrics

We implement two metrics to measure disentanglement of the latent variables. The "β-VAE metric" was proposed in [4] and uses the accuracy of a linear classifier as a disentanglement metric. The "Mutual Information Gap" (MIG) was introduced by Chen *et al*. [2]. Both metrics range from 0 (bad) to 1 (good). For our implementation of the metrics we used the details specified in [7].

	Question	Physics	Combined
β-VAE Metric	0.260	0.239	**0.719**
MIG	0.012	0.016	**0.252**

Table 1. β-VAE metric and MIG of the three trained models

Table 1 shows the results of both metrics on the three trained models. Our proposed Physics-VAE and the Question-VAE have similarly low scores in both metrics. Combining the two approaches however leads to a significant improvement.

6.2. Qualitative Results

Looking at qualitative results is a great way to evaluate disentanglement and reconstruction. We will not show visual results of the Physics-VAE because the task of predicting only the next frame is comparably easy and not well suited to evaluate disentanglement. Instead, we will focus on the two models using Physical Question Answering. Figure 7 shows a walk over every possible question for two example sequences next to the ground truth sequence. The combined VAE outperforms the Question-VAE both in its accuracy in following the trajectory and in reconstructing the correct shape of the ball.

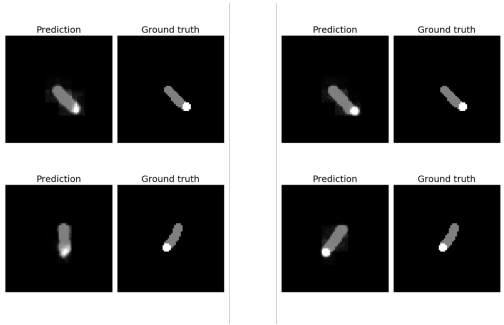

Figure 7. Complete walk over all possible questions of two example sequences compared to the ground truth sequence. Left: Question-VAE. Right: Combined VAE.

We also perform a walk over the latent variables. In order to do so, we calculate the latent encoding of the evaluation data set for every model. While holding five of the six variables constant at the mean of the observed values for one model, one latent variable is sampled between its observed minimum and maximum value. This shows the influence of one latent variable on the generated output image. We strongly recommend the reader to view these results on our Git `https://github.com/Mr-Pepe/dl4cv`.

7. Conclusion

We have shown that by combining the question approach of [5] with our proposed physics layer, it is possible to perform disentanglement of generative factors on video data. Our model successfully learned the position and velocity of the object, but failed to accurately predict its acceleration.

This however might also be due to a non-extensive manual hyperparameter tuning. Furthermore, Locatello *et al*. [7] point out that using disentangling autoencoders is highly sensitive to random initialization. An extensive analysis of hyperparameters and initializations is therefore left for future work.

References

[1] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner. Understanding disentangling in β-vae. *arXiv preprint arXiv:1804.03599*, 2018. 1

[2] R. T. Q. Chen, X. Li, R. Grosse, and D. Duvenaud. Isolating sources of disentanglement in variational autoencoders, 2018. 3

[3] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016. 2

[4] I. Higgins, L. Matthey, C. B. Arka Pal, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework, 2017. 1, 2, 3

[5] R. Iten, T. Metger, H. Wilming, L. del Rio, and R. Renner. Discovering physical concepts with neural networks, 07 2018. 1, 2, 3

[6] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2014. 2

[7] F. Locatello, S. Bauer, M. Lucic, G. Rtsch, S. Gelly, B. Schlkopf, and O. Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations, 2018. 3

[8] L. Matthey, I. Higgins, D. Hassabis, and A. Lerchner. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017. 1