ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

Complex Valued Gated Auto-encoder for Video Frame Prediction

Niloofar Azizi, Nils Wandel, Sven Behnke
(azizi,wandeln,behnke@ais.uni-bonn.de)
Bonn University, Computer Science Department,
Endenicher Allee 19a, 53115 Bonn, Germany

Abstract. In recent years, complex valued artificial neural networks have gained increasing interest as they allow neural networks to learn richer representations while potentially incorporating less parameters. Especially in the domain of computer graphics, many traditional operations rely heavily on computations in the complex domain, thus complex valued neural networks apply naturally.

In this paper, we perform frame predictions in video sequences using a complex valued gated auto-encoder. First, our method is motivated showing how the Fourier transform can be seen as the basis for translational operations. Then, we present how a complex neural network can learn such transformations and compare its performance and parameter efficiency to a real-valued gated auto-encoder. Furthermore, we show how extending both — the real and the complex valued — neural networks by using convolutional units can significantly improve prediction performance and parameter efficiency.

The networks are assessed on a moving noise and a bouncing ball dataset.

1 Introduction

Video prediction is the task of predicting future frames by extracting complex spatio-temporal features from a sequence of seed frames. In recent years Deep Neural Networks (DNNs) showed promising results in video prediction [1, 2].

Michalski et al. [3] proposed the Predictive Gating Pyramid (PGP) architecture to learn and predict the transformation in a sequence of frames. In PGP as well as its equivalent fully convolutional architecture [4], a layer of mapping units encodes transformation using a Gated AutoEncoder (GAE). The GAE is designed based on the assumption that two temporally consecutive frames can be interpreted as a linear transformation of one another. GAE was improved by Alain and Olivier [5] by going into complex domain. Recently the analysis of DNN architectures in complex domain raised attention as it makes the learning process faster [6] and the optimization process easier [7].

In this paper, we extend the GAE with tied input weights [5] to perform video frame prediction and propose a convolutional form which drastically reduces the number of model parameters while significantly improving the performance on a Bouncing Balls dataset.

2 Frame Prediction using Deconvolution

A motivating example shows how the transformation between two images that are translated copies can be calculated by deconvolution.

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

Let X_{t-1} be the first image and $X_t(x,y) = X_{t-1}(x-t_x, y-t_y)$ be the second image corresponding to X_{t-1} translated by t_x, t_y. Then, the transformation of $X_{t-1} \Rightarrow X_t$ can be seen as a convolution of X_{t-1} with a δ-function $\delta(x-t_x, y-t_y)$:

$$X_t(x,y) = \int X_{t-1}(\hat{x},\hat{y})\delta(x-t_x-\hat{x}, y-t_y-\hat{y})d\hat{x}d\hat{y} = X_{t-1}(x-t_x, y-t_y) \quad (1)$$

Thus in order to obtain $\delta(x,y)$, one can deconvolve X_t with X_{t-1}:

$$\delta(x-t_x, y-t_y) = \left(\mathcal{F}^{-1}\frac{(\mathcal{F}X_t)(u,v)}{(\mathcal{F}X_{t-1})(u,v)}\right)(x,y) \quad (2)$$

Here, $\mathcal{F}$ denotes the two dimensional Fourier transform and $\mathcal{F}^{-1}$ denotes the two dimensional inverse Fourier transform.

After having obtained the transformation δ, it can be used to extrapolate X_t and calculate X_{t+k}:

$$X_{t+k} = \mathcal{F}^{-1}\left((\mathcal{F}\delta)^k \cdot \mathcal{F}X_t\right) = \mathcal{F}^{-1}\left(\left(\frac{(\mathcal{F}X_t)(u,v)}{(\mathcal{F}X_{t-1})(u,v)}\right)^k \cdot \mathcal{F}X_t\right) \quad (3)$$

A schematic depiction of these operations can be found in Figure 1 a). While this works in theory, in practice usually several problems occur: the problem is ill posed, thus very sensitive to noise and in principle even multiple solutions could be obtained. $|\mathcal{F}X_{t-1}|$ can become arbitrarily small so one usually has to add a tiny offset ϵ in the denominator. Furthermore, this method makes the assumption of a periodic boundary and a uniform translation of the whole image.

Thus we want to train a model which can learn by itself more robust basis transformations that are not only suited for translations but can also handle for example rotations. Also, the method should be able to cope with multiple different local transformations arising from different objects in the scene.

3 Gated Auto-encoders

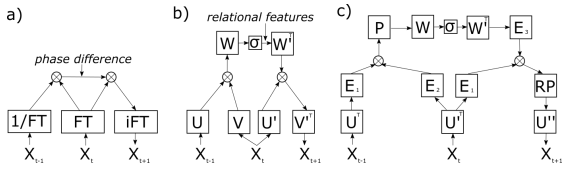

Fig. 1: a) Frame prediction using deconvolution, b) real valued GAE, c) complex valued GAE. U, U', U'', V, V', W, W' denote, depending on the version of the architecture, either fully connected or convolutional units. The weights of the real and complex valued GAE are shared among $(U, U', U''), (V, V')$. Sharing weights among (W, W') strongly decreased performance.

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

We refer to Gated Auto-Encoders [8] as "real valued GAE" and to Gated Auto-Encoders with tied input weights [5] as "complex valued GAE". If the transformation between images is linear, it can be written as:

$$X_t = LX_{t-1} \tag{4}$$

Here, X_t and X_{t-1} denote the vectorized form of the two images and L denotes a transformation matrix. If we further assume, that the transformation is orthogonal, L can be decomposed into:

$$L = UDU^* \tag{5}$$

with U being a unitary matrix ($UU^* = I$) and D being a diagonal matrix containing complex numbers of absolute value 1. While the assumption of a linear orthogonal transformation at first glance seems quite restricting it still comprises for example all transformations that can be described as pixel permutations (e.g. translation / rotation / shearing). From 4 and 5, it follows that:

$$U^*X_t = DU^*X_{t-1} \Rightarrow D = \text{diag}\left(\frac{U^*X_t}{U^*X_{t-1}}\right) \tag{6}$$

This is remarkable since it shows how orthogonal linear transformations can be represented in a much more compact way as rotations in the complex plane by D when the basis is properly changed by U. If we consider, for example, only translational transformations, U^* basically becomes a discrete Fourier transform and D corresponds to phase-differences in frequency domain. This corresponds exactly to what is described in Section 2 and again, the obtained representation of the transformation can be used to extrapolate to future frames:

$$X_{t+k} = UD^kU^*X_t \tag{7}$$

As for the introductory example, this leads to problems when projections of U lead to small absolute values, because in this case the computation of D becomes ill-conditioned resulting in falsely detected transformations.

Thus, the real valued GAE [8] as depicted in Figure 1 b) was developed. In this architecture, two separate trainable linear modules U and V learn representations of U^* and phase-shifted representations of U^*, respectively. Furthermore, in order to properly normalize the transformation representation, two additional linear modules (W and W'^T) and a sigmoid activation function are used. This representation then is either multiplied ("gated") by $U'X_{t-1}$ in the case of reconstruction or by $U'X_t$ in the case of prediction and projected back by V'^T:

$$X_t = V'^T(U'X_{t-1} \cdot W'^T\sigma(W(UX_{t-1} \cdot VX_t))) \qquad \text{reconstruction} \tag{8}$$

$$X_{t+1} = V'^T(U'X_t \cdot W'^T\sigma(W(UX_{t-1} \cdot VX_t))) \qquad \text{prediction} \tag{9}$$

This real valued GAE is able to learn robust relational features for a wide range of linear transformations [8]. However, [5] pointed out that parameter efficiency

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

can be drastically increased since U and V learn mostly the same features. A complex valued GAE was suggested, which directly makes use of Eq. 6. By treating U^* in the complex domain, it is not further needed to learn U and V separately. Instead, by carefully designing the matrices E_1, E_2, E_3, P, R (see [5] for exact definitions), the network is able to perform all computations directly in complex domain and neighboring weights in U now correspond to real and imaginary parts (see Fig. 2). This way, the network is able to spare out V of the real valued GAE (see also Figure 1 c for a schematic depiction) which not only results in fewer parameters but also in a strong prior potentially speeding up convergence. While [5] showed, how complex valued GAE can be used for reconstruction (see Eq. 10), we were able to evidence that it is as capable for prediction:

$$X_t = U'' R P(E_1 U'^T X_{t-1} \cdot E_3 W'^T \sigma(W P(E_1 U^T X_{t-1} \cdot E_2 U'^T X_t))) \quad \text{reconstruction} \tag{10}$$

$$X_{t+1} = U'' R P(E_1 U'^T X_t \cdot E_3 W'^T \sigma(W P(E_1 U^T X_{t-1} \cdot E_2 U'^T X_t))) \quad \text{prediction} \tag{11}$$

Since GAE scale quadratic in the number of image pixels, we also propose a convolutional form of the real and complex valued GAE which replaces the originally fully connected modules U, V, W by convolutional units. The matrices E_1, E_2, E_3, P, R for the complex valued GAE also stay the same but now are applied on the corresponding channels.

4 Experiments and Results

We performed several frame prediction experiments on different datasets in order to investigate the complex valued gated auto-encoder in the fully connected as well as in the convolutional setting. The network was trained to minimize the mean square error of the predicted frame with respect to the real follow-up frame.

	number of weights	Loss
real GAE, fc	270400	0.63
complex GAE, fc	155976	0.62

Table 1: MSE on the Moving Noise

	number of weights	Loss
real GAE, fc	1718400	3.3e-3
complex GAE, fc	903496	3.0e-3
real GAE, conv	5640	1.7e-4
complex GAE, conv	3241	2.1e-4
real PGP, 2 layers, conv	103240	3.8e-5
complex PGP, 2 layers, conv	52481	3.1e-5

Table 2: MSE on the Bouncing Ball

4.1 Moving Noise

The moving noise dataset consists of image sequences containing Gaussian noise which is either uniformly moved in a random direction or uniformly rotated by a random angle. The resolution is 24 x 24 pixels. Table 1 presents quantitative results and Figure 2 shows the weights in U learned by the model after convergence. They visually look very similar to the weights obtained by [5], confirming

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

that complex GAE can be used for prediction as well. Table 1 shows that the complex valued GAE performs slightly better than the real valued GAE while incorporating significantly fewer weights.

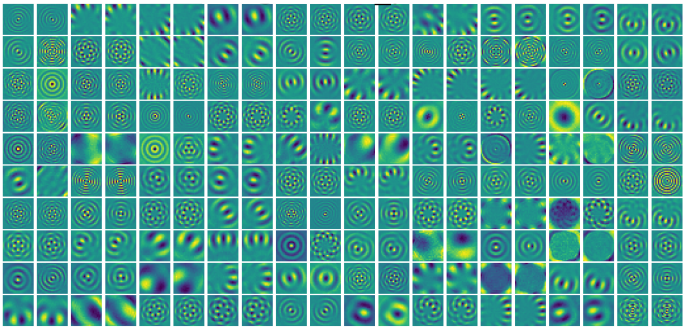

Fig. 2: weights of U learned by the complex valued fully connected GAE. The pairing of neighboring real and imaginary parts is clearly visible.

4.2 Bouncing Balls

This dataset consists of 2 black balls that uniformly move in random directions of the 2D image plane. If a ball hits a wall or another ball, its movement gets reflected. The resolution is 64 x 64 pixels. Figure 3 gives an idea about the qualitative results obtained using the convolutional complex valued GAE.

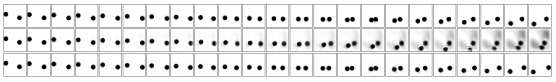

Fig. 3: top row: ground truth, 2nd row: complex valued GAE, 3rd row: complex valued PGP. First 3 frames: seed frames, remaining 20 frames: predictions. In this example, the resolution is only 32 x 32 pixels and the training loss was averaged over 3 subsequent predicted frames to train the network on longer time horizons.

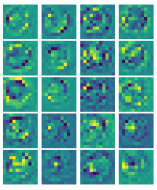

Fig. 4: weights of U learned by the complex convolutional GAE

Since both balls move in different directions, the space of possible transformations is huge. This makes the problem especially hard for a fully connected network. If we however use convolutional units whose kernels only cover a small part of the image, the transformations again correspond only to translations and can be more easily learned, while the number of parameters is drastically reduced. Our experiments (see Table 2) support this claim. Figure 4 clearly shows, that the convolutional model learns kernels that are able to shift a ball

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

in different directions. Interactions between balls however cannot be modelled by this class of GAE as they violate the linearity assumption we made in the beginning:

$$X_t^{\text{ball}_1+\text{ball}_2} = L(X_{t-1}^{\text{ball}_1} + X_{t-1}^{\text{ball}_2}) \neq LX_{t-1}^{\text{ball}_1} + LX_{t-1}^{\text{ball}_2} = X_t^{\text{ball}_1} + X_t^{\text{ball}_2} \quad (12)$$

To deal with such cases, one has to also include higher order transformations - for example as shown by Michalski et al. [3] with Predictive Gating Pyramids (PGP), which we refer to in the following as real valued PGP. A complex valued PGP can be obtained by replacing all real valued GAE inside the real valued PGP architecture by complex valued GAE. Experiments with real and complex valued PGP are also reported in Table 2. Qualitative results (see Figure 3) indeed show superior performance of the complex valued PGP over the complex valued GAE when interactions happen.

5 Conclusion

In this work, we first showed, how the notoriously unstable deconvolution operation fits into the framework of gated auto-encoders. We then presented a way of extending complex valued GAE to perform predictions. Furthermore, we demonstrated that the complex convolutional form is more efficient than the complex fully connected form on the bouncing ball dataset. Our work puts some foundations on complex valued convolutional gated auto-encoders and closes the loop between real valued GAE [8] and complex valued GAE [5] and PGP.

Acknowledgment This work was funded by grant BE 2556/16-1 (Research Unit FOR 2535 Anticipating Human Behavior) of the German Research Foundation (DFG).

References

[1] S. Xingjian, Z. Chen, H. Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems*, 2015.

[2] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

[3] V. Michalski, R. Memisevic, and K. Konda. Modeling deep temporal dependencies with recurrent grammar cells"". In *Advances in Neural Information Processing Systems*, 2014.

[4] Filip de Roos. Modeling spatiotemporal information with convolutional gated networks. Master's thesis, Chalmers Institute of Technology.

[5] Droniou Alain and Sigaud Olivier. Gated autoencoders with tied input weights. International Conference on Machine Learning, 2013.

[6] Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. In *International Conference on Machine Learning*, 2016.

[7] T Nitta. On the critical points of the complex-valued neural network. In *Neural Information Processing (ICONIP)*. IEEE, 2002.

[8] Roland Memisevic. Learning to relate images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.