3M2-2

Video Compression with a Predictive Neural Network

Lana Sinapayen*1 Takashi Ikegami*1

*1 The University of Tokyo

Predictive networks are a type of generative neural network model that learns to minimize the error between predicted data and real input. Prediction is used as a way to perform unsupervised learning of latent structure in the data, for example shapes and linear transformations in images. As a result, video-trained predictive networks can produce output by processing input through intrinsically stored invariances. In this study we propose to use such learned invariances as a compression/ decompression engine for videos on spatial and temporal dimensions.

1. Introduction

After being first used for classification purposes, Deep Learning is now starting to be used in a new area called Predictive Coding [Kanai 2015]. The main idea is to use a Neural Network to predict time series, instead of directly classifying the contents of time series. This is particularly effective to perform unsupervised learning, as these networks have been shown to learn pattern recognition as a side effect of the prediction task [Lotter 2015].

On the other hand, the most common way of encoding times series of images, the MP4 format [Wiedegand 2003], relies on a relatively simple prediction engine calculating Motion Vectors (Fig. 1).

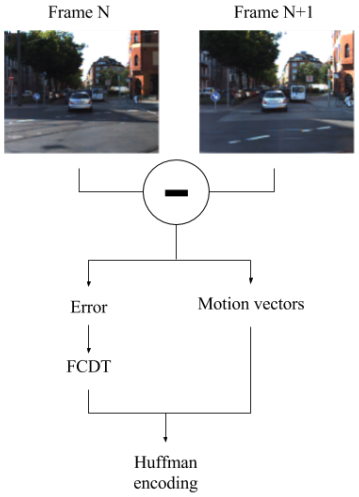

Fig. 1: Basis of MPEG4 encoding. The Motion Vectors technique is used to predict the next frame, and the resulting error is encoded into the video file for future correction.

2. Proposal

We propose to use the PredNet [Lotter 2016] architecture, an high performance Predictive Neural Network, as a prediction engine for video compression (Fig. 2). One of our goals is to replace the prediction engine in video compression algorithms (for example, in the MP4 format) to obtain higher compression rates.

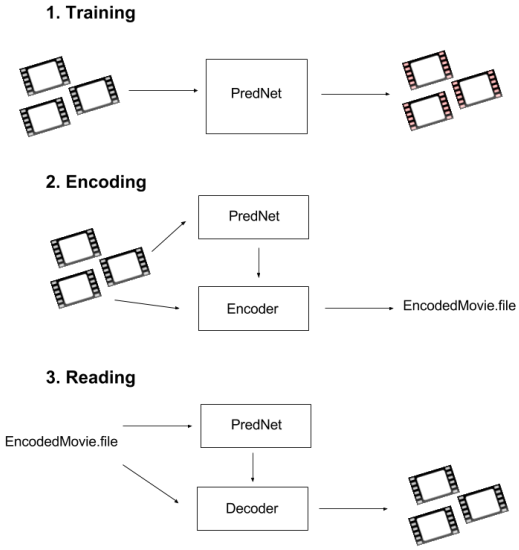

Fig. 2: Proposed encoding/decoding method. The PredNet model is trained on sample videos. Then the trained model is used to predict frames in the video we want to encode, and the prediction error is encoded the same way as MPEG4.

PredNet works by training successive macro-layers of the network to predict the error on the output of the previous macro-layer. Each macro-ayer contains 4 sub-networks: the recurrent network that contains prediction representations, the networks computing the prediction of the input at t+1, the network representing the real input at t+1, and network calculating the error term from comparing input and prediction (Fig. 3).

2.1. Encoding

Here we use a pre-trained version of PredNet to encode one of the videos of the KITTI dataset [Geiger 2013]. The network is already trained and the internal weights are fixed and not updated during use. The network has never been trained on the video we encode. Instead of running the network on a full batch of data, we use the first image of a video as input at time = 0 frames. The prediction at t+1 by PredNet is then compared to the actual frame

Contact: Lana Sinapayen lana@sacral.c.u-tokyo.ac.jp

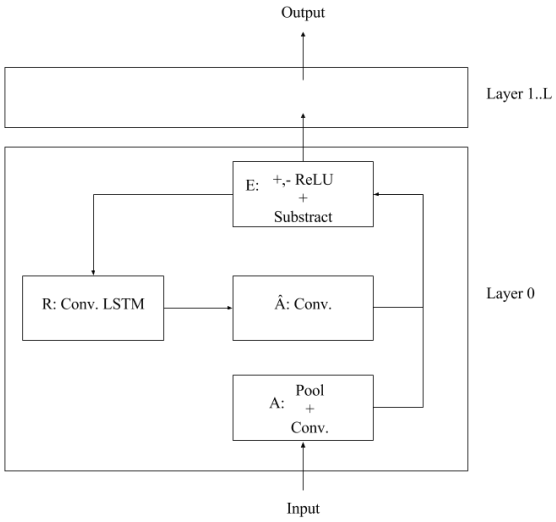

Fig. 3: PredNet architecture. The transformed input from A is compared with the prediction from R, and the resulting error calculated by E is sent back to R and also serves as input A for the upper layer L+1.

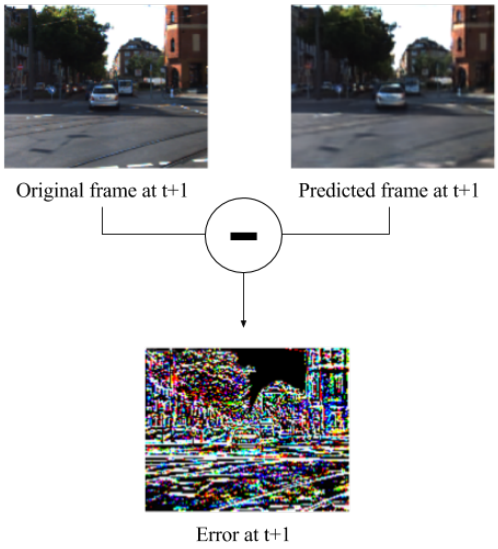

Fig. 4: Encoding. The predicted frame and the original frame are compared, and the error between the two is saved. This is repeated for each frame of the video. The encoded movie file therefore only contains the error information.

at t+1, and the resulting error is recorded in a separate file (Fig. 4). The process is repeated for the whole video. We obtain a file containing all the errors, that is all the information necessary to correct the predicted video back to the original video.

2.2. Decoding

In the decoding phase, the first image of the video is given as input to PredNet. The resulting prediction is corrected using the error file recorded during the encoding phase. This image is exactly equivalent to the image of the original video at t+1. That image is then itself used as input for PredNet, and the process is repeated until the end of the video (Fig. 5).

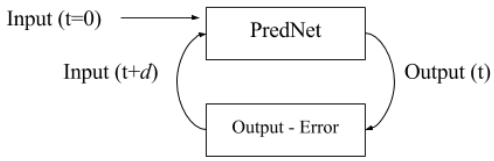

Fig. 5: Decoding. The predicted output at t is corrected using the known error previously encoded in the encoding phase. This corrected output is then used as input for the next timestep. The whole video can be reconstructed frame by frame this way.

3. Discussion

We confirmed that our proposed encoding method allows for lossless encoding of videos. As a prediction engine, PredNet gives better results than the basic prediction engine used by the MP4 standard, therefore once integrated to the MP4 algorithm it should give much better compression rates than the original MP4. This compression rate will vary depending on the contents of the video and how “predictable” it is.

Our proposed method still has disadvantages: the engine is not universal and must be trained before encoding, therefore the compression rate will depend on the type of movie being encoded. The network is optimised to run using batch processing used in other Deep Learning networks, not for continuous runs, therefore it can be slow on a standard computer.

As a future work, we propose to add time compression to our method, by monitoring the error size during the encoding phase. Instead of recording the error at each timestep, we would record the error only when it is above a threshold. During the steps in between, the network can just run on its own uncorrected predictions.

References

R. Kanai, Y. Komura, S. Shipp, and K. Friston. Cerebral hierarchies : predictive processing, precision and the pulvinar. Philos Trans R Soc Lond B Biol Sci, 2015.

T. Wiegand, G. J. Sullivan, G. Bjontegaard, A. Luthra, Overview of the H. 264/AVC video coding standard. IEEE Transactions on circuits and systems for video technology, 13(7), 560-576, 2003.

W. Lotter, G. Kreiman, D. Cox, Unsupervised learning of visual structure using predictive generative networks. arXiv preprint arXiv:1511.06380, 2015.

W. Lotter, G. Kreiman, D. Cox, Deep predictive coding networks for video prediction and unsupervised learning. arXiv preprint arXiv:1605.08104, 2016.

A. Geiger, P. Lenz, C. Stiller, R. Urtasun. Vision meets robotics: The KITTI dataset. International Journal of Robotics Research (IJRR), 2013.