Full Resolution Video Compression Using Recurrent Convolutional Neural Network

Berk Çoker, Cedric Yue Sik Kin
Stanford University

Introduction

From Netflix to Youtube, most of the platforms we now use are dominated by video content. According to a Cisco study, video traffic will become 82 percent of the entire consumer internet traffic by 2019 [1]. These predictions highlight the importance of creating novel ways of transferring and storing video content.

Consumer Internet Video 2014–2019							
	2014	2015	2016	2017	2018	2019	CAGR 2014–2019
By Network (PB per Month)							
Fixed	20,485	25,452	32,981	43,226	56,771	74,319	29%
Mobile	1,139	2,014	3,475	5,842	9,407	14,999	67%
By Category (PB per Month)							
Video	18,437	22,940	30,242	40,907	55,931	76,771	33%
Internet video to TV	3,188	4,526	6,214	8,160	10,248	12,548	32%

Figure: Detailed predictions about consumer video consumption from Cisco study

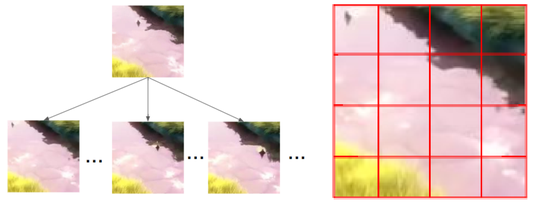
ConvNets have become a prevalent tool in high-level computer vision tasks like image classification or object detection. But recent work has shown their relevance to low-level image and video processing tasks. Google [2] recently showed an autoencoder architecture capable of compressing and reconstructing full resolution images with performances on par with standard image compression algorithms such as jpeg while Kapperler et al. [3] proposed an architecture that is trained on both the spatial and temporal dimensions of video frames to enhance resolution.

Problem Statement

Given a video of arbitrary dimensions and size, we propose a model that aims to obtain a noise-free and blur-free compressed version of the video with variable compression ratios available.

Dataset

We overfit the model by training it on a particular video.



(a) Extract frames from video into a lossless format like png

(b) Split frames into 32x32x3 quadrants

Figure: Pipeline for preprocessing dataset

Method

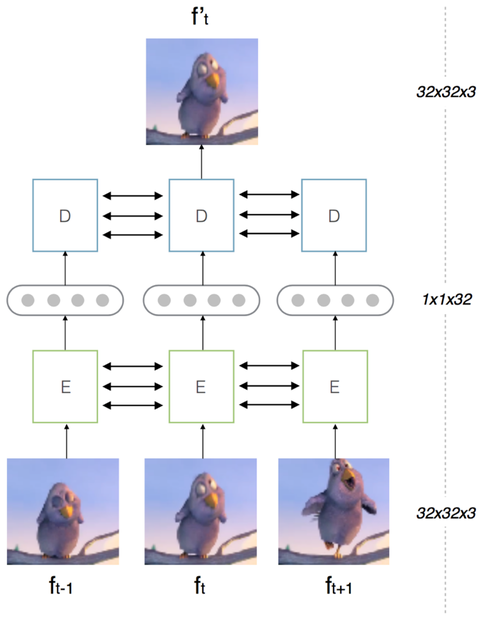


Figure: Our architecture. We exploit the temporal dependencies between frames.

Our network consists of encoders that reduces the dimensionality of the frames, and decoders that estimates the original frame; both the encoder and decoder consist of gated recurrent units. The formulation for GRU, with input x_t and hidden state/output h_t is:

$$z_t = \sigma(W_z x_t + U_z h_{t-1})$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1})$$

$$h_t = (1 - z_t) \circledast h_{t-1} + z_t \circledast \tanh(W x_t + U(r \circledast h_{t-1}))$$

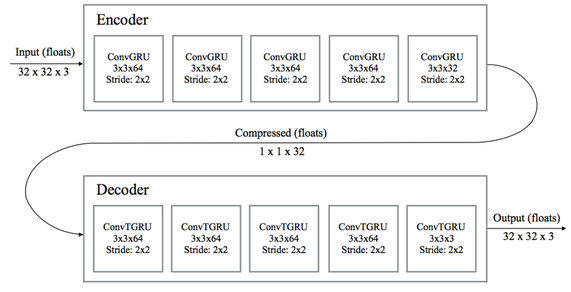


Figure: Our architecture. We exploit the temporal dependencies between frames.

Experimental Evaluation

For evaluation, we use the most common evaluation metrics to measure image quality: Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measurement (SSIM).

$$\text{PSNR } (\hat{Y}, Y) = 20 \log(s) - 10 \log \mathbf{MSE}(\hat{Y}, Y)$$

$$\text{SSIM } (\hat{Y}, Y) = \frac{(2\mu_{\hat{Y}}\mu_Y + c_1)(2\sigma_{\hat{Y}Y} + c_2)}{(\mu_{\hat{Y}}^2 + \mu_Y^2 + c_1)(\sigma_{\hat{Y}}^2 + \sigma_Y^2 + c_2)}$$

where s is the maximum possible pixel value (255 in our case), μ_Y denotes the mean of image Y, μ_Y^2 the variance, $\sigma_{\hat{Y}Y}$ the covariance of the two images and c_1, c_2 typically set to $0.01s^2$ and $0.03s^2$ respectively.
We then take the average among all the frames to get the final value.

	PSNR	SSIM
JPEG	37.5847 dB	0.9836
3-RCNN (1 Time Step, Window Size 7)	37.5971 dB	0.9883
3-RCNN (3 Time Step, Window Size 7)	38.8007 dB	0.9910

Table: Our results for 8x compression rates.

(a) Compressed frame in JPEG format (8x Compression)

(b) Reconstructed frame with 3-CRNN, 1 Time Step, Window Size 7

(c) Reconstructed frame with 3-CRNN, 3 Time Step, Window Size 7

Figure: Visual Comparison between JPEG and our model's outputs

Conclusion

Given the results from the evaluation metrics, we are able to exploit the intrinsic temporal dependencies between video frames by considering neighboring frames when predicting a video frame. We are able to reconstruct video frames from a compressed size with better performance compared to JPEG compression.
The given results were based on a toy dataset. For future work, we'll focus on extending the model's capacity to reconstruct full video frames and investigate the usefulness of this technique to applications like security footage.

[1] Cisco. White paper: Cisco vni forecast and methodology,2015-2020, 2016. [2] G. Toderici et al. Full resolution image compression with recurrent neural networks, 2016.

[3] A. Kapperler et al. Super-resolution of compressed videos using convolutional neural networks. International Conference on Image Processing, 2016.