Data Compression Platform

Malkan, Ishan Hitesh
Parmar, Mithil Ashish
Thaker, Jayam
Trivedi, Yash Jigneshkumar

- With easy access to high quality cameras with everyone, the rise of video data is significant.
- Storing and sharing large amounts of such videos require expensive storage and network facilities
- To solve this problem we come up with a video compression platform that can be used and improved upon by an open source community.

Research

Before directly jumping on to the project, we performed an in depth research of the existing video compression algorithms of various categories as mentioned here:

- Lossless compression
- Lossy compression
- Traditional compression
- ML-based compression

Methodology

Our Approach

1. We narrowed down our research to ML-based approaches after some exploration.

2. We focused on Super Resolution Generative Adversarial Networks (SR-GAN) for our compression.

3. SR-GAN's architecture is same as a GAN as shown in the image.

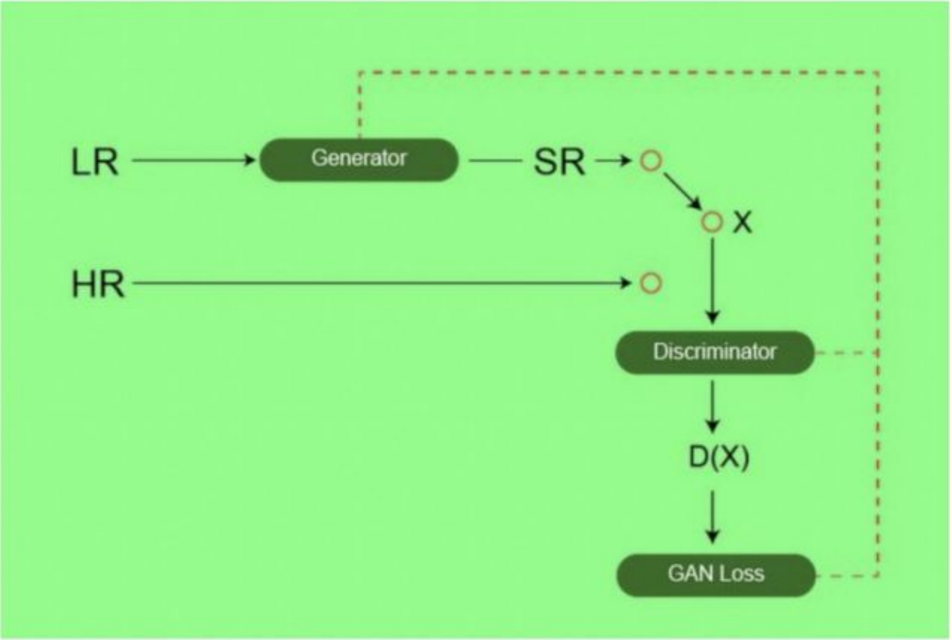
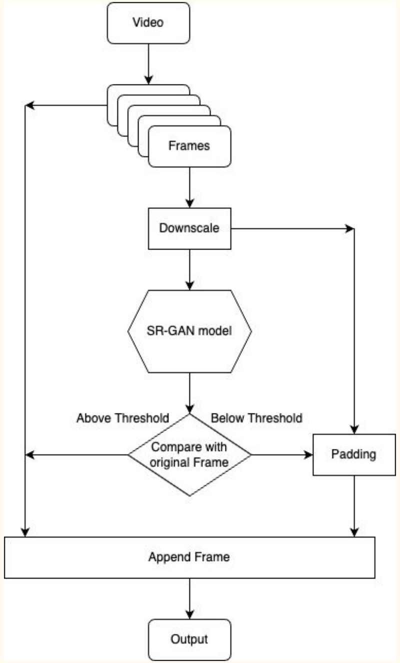
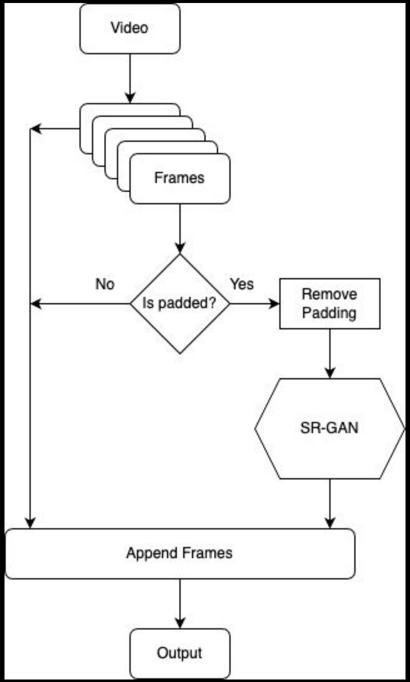


Fig. 1. SR-GAN Architecture

We designed our compressor where each frame is downscaled and again up-scaled using SR-GAN, if the difference is less than a threshold we keep the frame downscaled else we keep it as it is. Architecture is as the follows:



We designed our decompressor where each frame is checked whether it is downscaled or not and for the ones that are, we upscale them again using SR-GAN and join with the others. Architecture is as the follows:



Evaluation and Results

- To evaluate the decrease in video size, we used a modified version of compression ratio (CR) that gives us the ratio between the new size of the video compared with the old size. The formula for the same is as follows:

$$CR' = (h1 \times w1 \times b1 \times f1) \div (((h1 \times c) \times (w2 \times c) \times b2 \times f2 \times n) + (h1 \times w1 \times b2 \times f2 \times (t - n)))$$

- Here h1 is the height of each frame of the original video, w1 is the width of each frame of the original video, b1 is the bits per pixel in each original video frame, and f1 is the number of frames per second in the original video. Similar h2, w2, b2, and f2 are the same things for the compressed video. n means the number of frames that are downscaled, then t is the total number of frames, and c is the ratio of downscaling

- To evaluate the reduction in quality of a frame, we used the Root Mean Squared Error (RMSE) between the element wise pixels, and averaged the results for each frame to get the same for video. The formula for the same is as follows:

$$RMSE(frame) = \sqrt{\left[\left\{\sum_{i=1}^{h1}\sum_{j=1}^{w1}\sum_{k=1}^{3}\left(f1_{ijk} - f2_{ijk}\right)^2\right\} \div \{h1 * w1 * 3\}\right]}$$

- Here, f1ijk is the normalized value of the kth color value of the j th column of the ith row pixel, and similarly for f2ijk

Results

For the frame level, we observed that on average we were able to beat the OpenCV library's resize method by around 5% using our RMSE formula mentioned above. For the video level, if all the frames are compressed, we tested with several videos, and the RMSE ranged from 0.1 to 0.2 and averaged near 0.16. Hence with an average of 0.16 RMSE, we were able to reduce the video size by 4 times (the model we used performs 4x super-resolution as mentioned in one of the previous sections), and when we set the threshold value as the median of the RMSE values of each frame, our average RMSE reached 0.07 and the video size became $0.5 + (0.5*0.25) = 0.625$ (62.5%) of the original size. Playing around with these values gave us a general idea of how the model is performing and what areas require more training.

Conclusion

In conclusion, as discussed in the previous sections, this journey of video compression led us to explore a lot of lossless and lossy methods, as well as traditional and ML-based algorithms. Finally, we were delighted to find out that our approach of SR-GAN-based compression beat the OpenCV algorithms and gave users much more flexibility in handling the compression vs loss tradeoff. Finally, we have planned several enhancements to this project that can start off a community platform.

Thank You!