111 學年度第 1 學期
Video Processing

Final Project:

Enhanced Deep Video Compression in Feature Space

Students：　柯佐勳、林佑珊、Manu

Department：Electrical Engineering Institute

Professor：　江瑞秋 老師

Date： 2023.1.13

Abstract

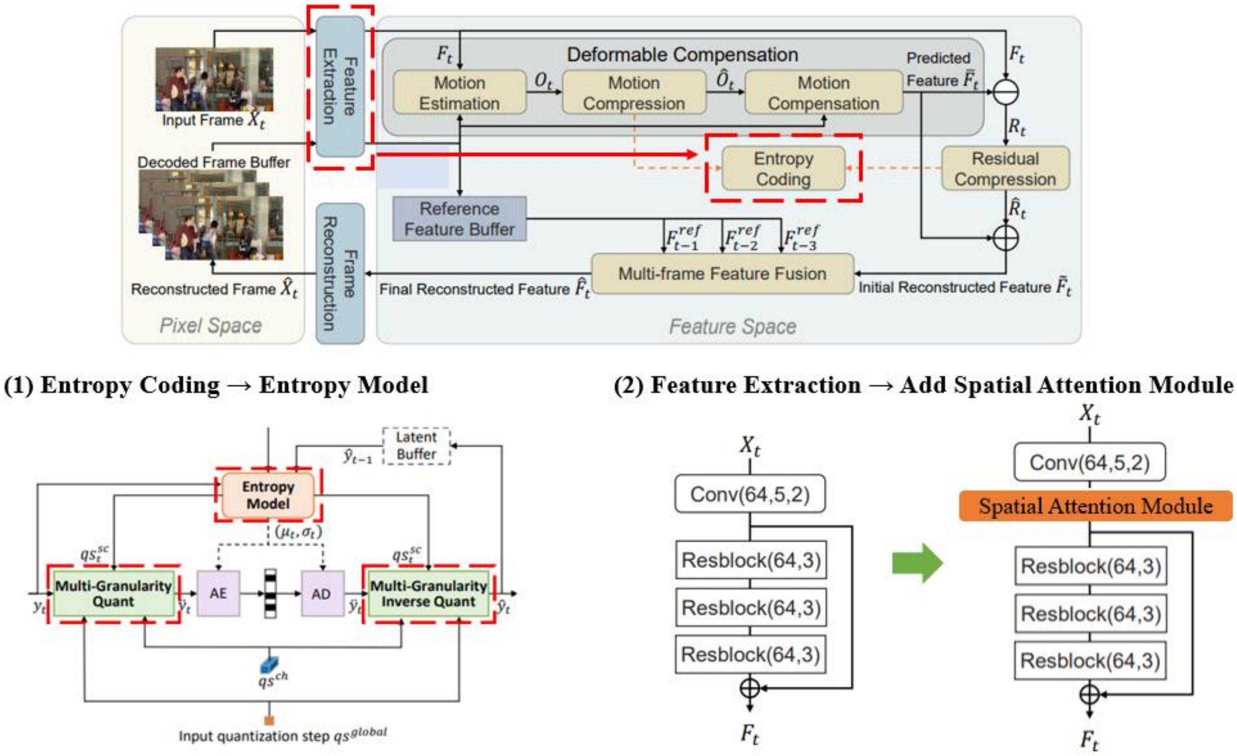
This research project, aims to improve the well known FVC video coding network, by enhancing it’s capabilities with different deep video coding techniques mainly presented in DCVC, but not only. The goal of the report is not to focus on the results, but rather to create a deeper understanding of the learnable video concepts, along with improving the technical abilities of the students.

1. Introduction

Recent deep video compression work [1,2] has achieved impressive results by applying deep neural networks in a hybrid video compression framework. Currently, most works only rely on pixel-level operations (e.g., motion estimation or motion compensation) to reduce redundancy. However, it is difficult to generate accurate pixel-level optical flow information for videos with complex non-rigid motion patterns at the pixel level. Therefore, given the robust representation capabilities of deep features for various applications, it is desirable to perform motion compensation or residual compression in the feature space to more effectively reduce and synthesize the spatial or temporal redundancy in individual frames.

So in this research, our baseline is Feature Space Video Coding Network (FVC for short) which succeeded in reducing the spatio-temporal redundancy in the feature space[3]. This video compression-based method performs all operations in feature space for more accurate motion estimation, motion compensation, residual coding, entropy coding and quantization, where it can seamlessly incorporate deformable convolutions into a hybrid deep video compression framework. As a result, we can alleviate the errors introduced by inaccurate pixel-level operations like motion estimation/compensation and achieve better video compression performance.

Furthermore, our work aims to modify two parts based on the baseline FVC network. The **first part** is to replace the Entropy coding network based on the reference, and generates quantization steps on spatial channels [4], which can not only effectively capture spatial and temporal dependencies, but also helps our codec achieve the smooth rate adjustment in single model but also improves the final rate-distortion performance by dynamic bit allocation. The **second part,** is a necessity, to add the Spatial Attention Module to the Feature Extraction model. As it is used to help the entropy model to extract the spatial feature [5], as shown in Fig. 1. Finally, in the experimental chapter 4, the degree of optimization is compared with the ablation study, and the results of the first part and the second part of the baseline and the modification will be used as experimental verification.



(1) Entropy Coding → Entropy Model

(2) Feature Extraction → Add Spatial Attention Module

Figure 1. The illustration of the two parts of the modified FVC network

2. Related work

This chapter will briefly introduce the improvement and optimization of this project using the model frameworks of the two papers.

2.1 Hybrid Spatial-Temporal Entropy Modeling for Neural Video Compression [4]

This paper proposes a comprehensive **entropy model** which can efficiently leveraging both spatial and temporal correlations, As shown in Fig. 2. and then helps the neural video codec outperform the latest traditional standard H.266. In their entropy model, the dual spatial prior is proposed to reduce the spatial redundancy. Most existing neural codecs rely on the auto-regressive prior [5] to explore the spatial correlation. For neural video codec, another challenge is how to achieve smooth rate adjustment in single model.

However, most neural codecs lack such capability and use fixed quantization step (QS). To achieve different rates, the codec needs to be retrained. It brings huge training and model storage burden. To solve this problem, they introduce an adaptive **quantization mechanism at multi-granularity levels** ,as shown in Fig. 2. which is powered by their entropy. In addition, it is noted that using entropy model to learn the QS not only helps our codec obtain the capability of smooth rate adjustment in single model but also improves the final RD performance.

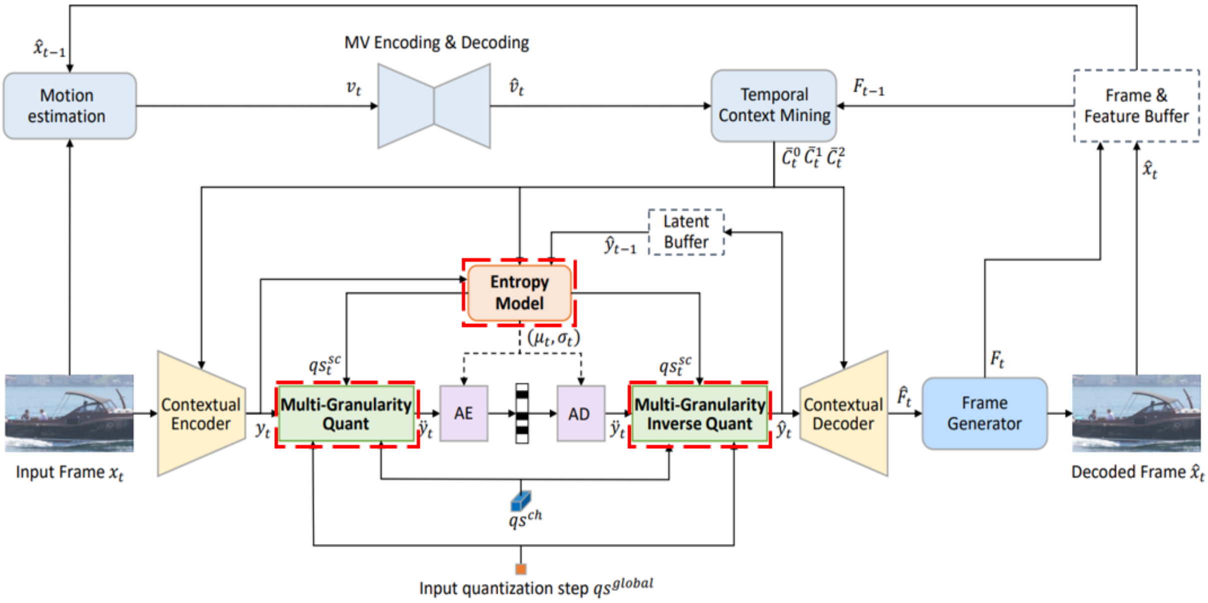


Figure 2. The illustration of Hybrid Spatial-Temporal Entropy Modeling for Neural Video Compression framework

(The red dashed box is the module we want to use in this project)

2.2 Spatial Attention Model [5]

This paper proposed a network module called Convolutional Block Attention Module (CBAM). CBAM contains two sequential sub-modules including channel attention module and spatial attention module, as shown in Fig. 3. But in our method, it has been decided to only use the spatial attention module.

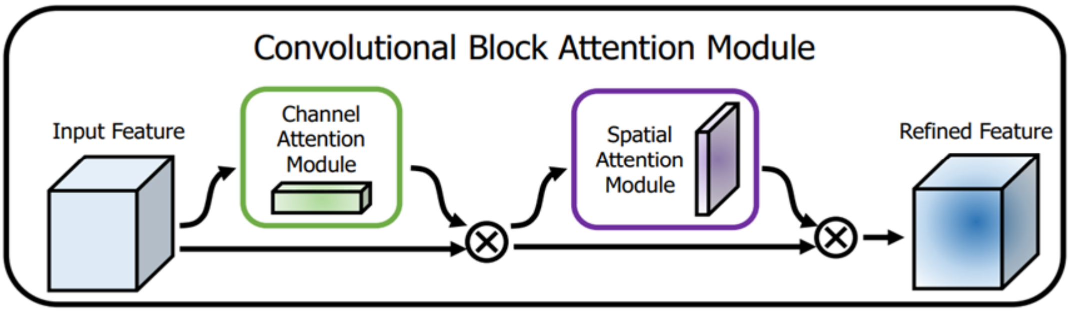


Figure. 3 Convolutional Block Attention Module [5]

Spatial attention module structure is illustrated in Fig. 4. In order to compute the spatial attention, they operate max-pooling and an average-pooling along the channel axis to get two 2D maps. Each map indicates average-pooled features and max-pooled features across the channel. Then these maps are concatenated and become a feature descriptor. On this feature descriptor, they add a convolution layer to produce a spatial attention map which contains spatial information.

In our modify method, we add this spatial attention module in the feature extraction module of our baseline to extract spatial information.

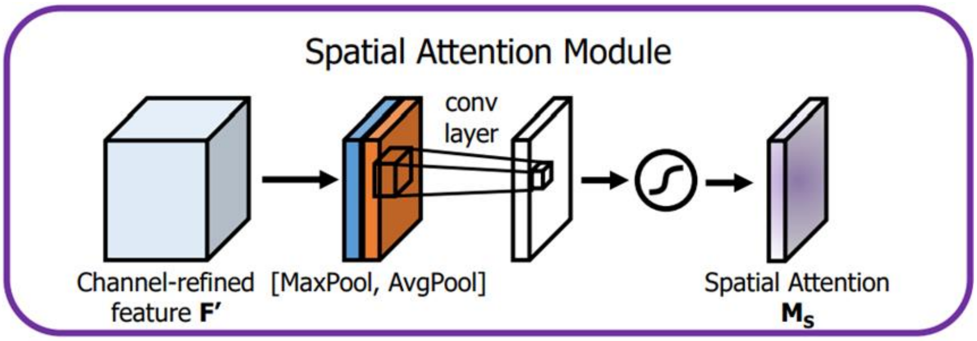


Figure. 4 Spatial Attention Module [5]

3. Methodology

This chapter will explain the method we will use to modified the FVC framework and the application of the module for analysis. The Hybrid Spatial-Temporal Entropy Model is in Chapter 3.1, and the Spatial Attention Module is in Chapter 3.2.

3.1 Hybrid Spatial-Temporal Entropy Model [4]

The entropy model in reference [4] gave us great inspiration and help, which can effectively capture spatial and temporal dependencies and generate quantization steps in terms of spatial channels. They introduce an adaptive quantization mechanism at multigranularity levels, which is powered by their entropy mode. In addition, it is worth noting that using the multi-granularity quantization provided by the entropy model to learn the quantization step (QS), as shown in Fig. 5. The quantized features from the motion compression and residual compression modules will be converted to a bitstream by performing entropy coding. And input feature F_t into Entropy Model together, and generate Spatial channel wise quantization steps qs^{sc}.

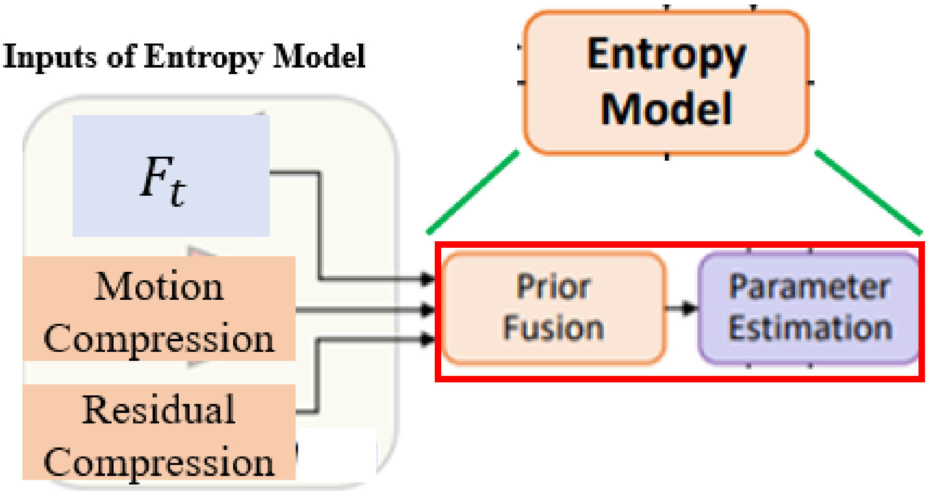


Figure 5. The illustration of the design and expected application of the Entropy Model of our project

Their **multi-granularity quantization** involves three different kinds of quantization step (QS): the global QS qs^{global}, the channel-wise QS qs^{ch}, and the spatial-channel-wise QS qs^{sc}, which not only helps the codec to obtain smooth rate adjustment in a single model, also improved final RD performance. The qs^{global} is only a single value and is set from the user input for controlling the target rate, As all positions take the same QS ,the qs^{global} brings a coarse quantization effect. Thus design a modulator qs^{ch} to scale the QS at different channels because different channels carry information with different importance. The qs^{sc} is generated by the entropy model, as shown in Fig. 6. for each frame, it is dynamically changed to adapt the video contents. Such design not only helps us achieve smooth rate adjustment but also improves the final RD performance by content-adaptive bit allocation.

In order to make the entropy model of our project model learn to allocate more bits to more important content, we use the Entropy Model in the paper [4] to modify the Entropy Coding in our baseline FVC framework[3], hoping to pass through our modified network not only does it take fullly exploits the spatial-temporal correlation, but also helps us achieve smooth rate adjustments, while also taking into account that the content is critical for the reconstruction of the current and subsequent frames, enabling the dynamic bit allocation of the adaptive quantization mechanism to improve final compression compare.

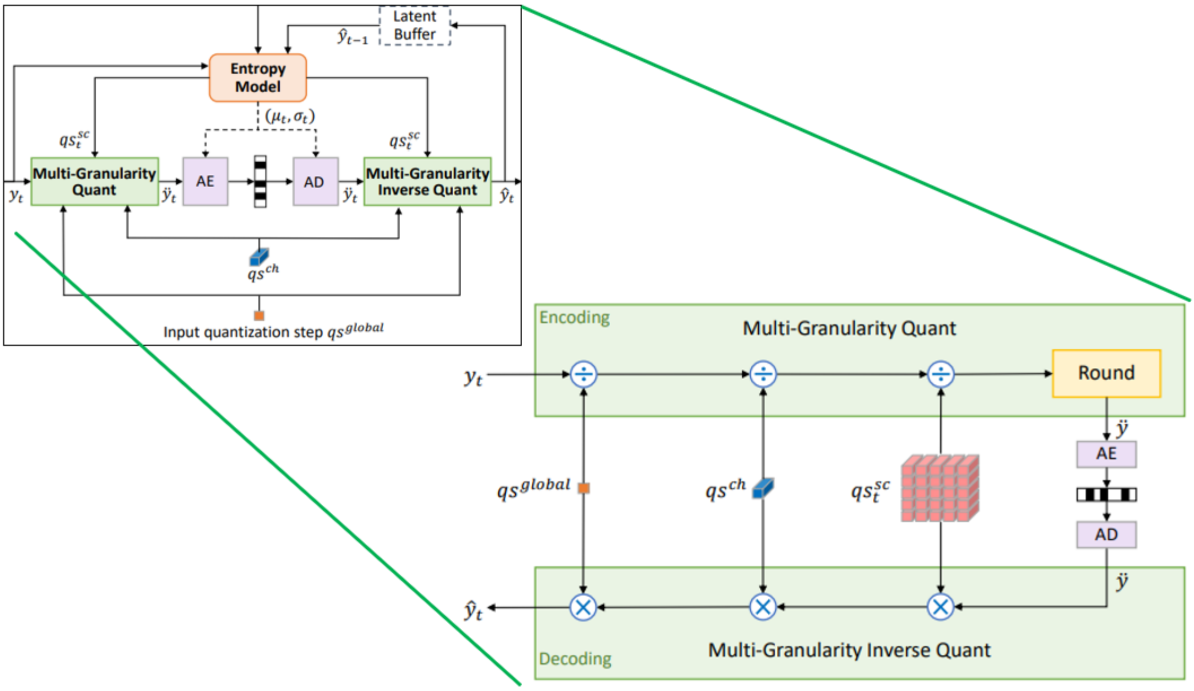


Figure 6. The illustration of entropy model , multi-granularity quantization and the corresponding inverse quantization

3.2 Spatial Attentation Model [5]

In our method, we replace the original entropy model with the Hybrid Spatial-Temporal Entropy Model. Compared to the original one, Hybrid Spatial-Temporal Entropy Model input not only the temporal information but also the spatial information. However, there is no spatial feature extraction in the FVC framework [3]. In order to obtain the spatial feature, we add a Spatial Attention Module from CBAM [5] to the feature extraction part in the baseline, as seen in Fig.7.

With our modification, the initial input frame and the preceding reconstructed frame are encoded as the feature to create the representations in the feature space. The feature extraction module creates the feature representation using a convolution layer with a stride of 2, which is followed by a spatial attention module and a number of residual blocks.

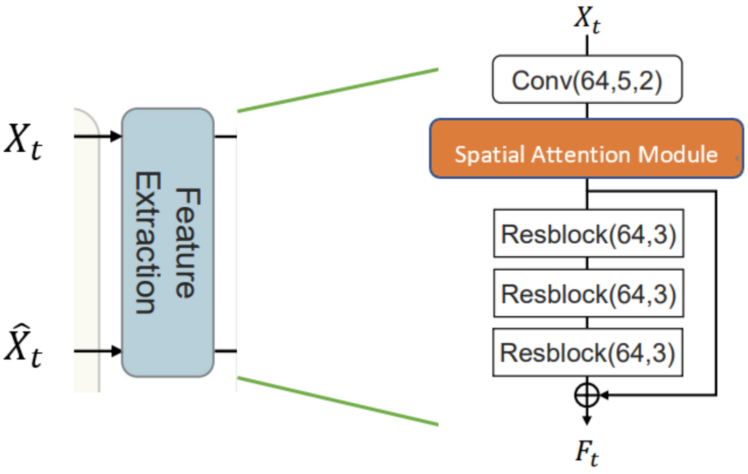


Figure 7. Feature extraction with a Spatial Attention Module

4. Experimental Results and Analysis

This section will present in detail the results of the practical implementations of the theoretical approach presented in section 3. The emphasis on this part will be a fair comparison between baseline and enhancement of it, for this work to truly conclude if there is actually benefit to the proposed add-ons.

4.1 Experimental Setup

For the training stage of all 3 models, the Vimeo-90k dataset was used as the training dataset. This dataset contains a total of 89,800 samples of video clips, each of which consists of 7 frames of resolution 488x256. As a data augmentation technique, a simple 256x256 random crop was applied to the frames before the training process began.

The testing dataset for this study was limited to a dataset of video sequences from the UVG database due to time constraints. The UVG dataset used in the study consisted of seven high frame rate videos (50, 60, and 120 frames per second) with resolutions of 1920x1080 and 3840x2160. These video sequences were chosen as they had small differences between consecutive frames.

The evaluation metrics used in this study included bpp (bits per pixel) to measure the average number of bits used for motion coding and residual coding for one pixel in each frame, as well as PSNR and MS-SSIM to evaluate the distortion between the reconstructed frames and the ground-truth frames. The PSNR and MS-SSIM for each video sequence was obtained by averaging the PSNR and MS-SSIM values for all frames.

As implementation details, all three models were trained differently with only one value for **λ = 256** using a simple one-stage training scheme. **Each model was trained for 3 epochs of the dataset with a learning rate of 1e-4.** The learning rate has been increased due to the lower number of training steps, so hopefully it will work as a compensation, and the total loss of performance will be decreased by this way.

Lastly, the remaining technical details, both hardware and software, are presented in Table 1.

Table. 1 Experimental setup for this project

Operating System	Ubuntu 20.04
CPU	Intel(R) Xeon(R) CPU E3-1245 v6 @ 3.70GHz
GPU	NVIDIA GeForce GTX 1080Ti
RAM	32GB
Open source libraries	PyTorch 1.13.1, Compressai 1.2.3
Programming Language	Python 3.8

4.1 Experimental Setup

In this section, there will be provided the experimental results, to fairly demonstrate the effectiveness of the proposed changes to the baseline FVC[3]. In Figure 8 there are presented the testing values obtained from the all three tested models. This will be a display of both the results and the ablation test, as it shows the best performance, and how the enchanting modules differently impact the overall performance of the final model, as shown in Fig. 8.

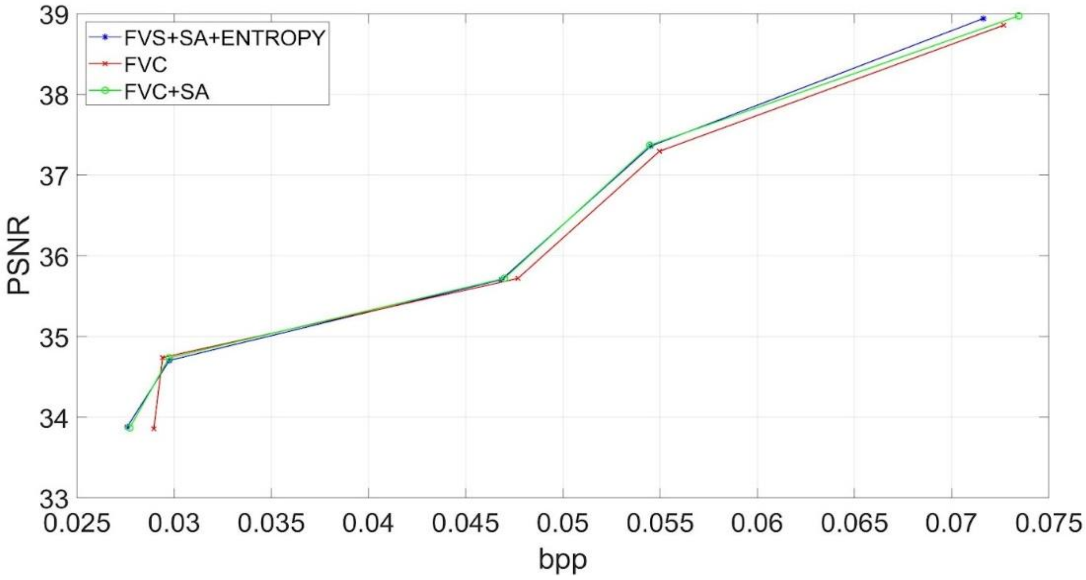


Figure 8. All models result on UVG training dataset, after 3 epochs.

It is clear to see that the models add a significant amount of improvements, especially in the higher bitrate. The FVC baseline model still underperforms compared to the initial published results, as a consequence of the small amount of training. In the original report, the training of the algorithm took 4 days, 3 days and 10h for all three individual steps of training (so a total of 7 days and 10h for only one λ value), so naturally because there was no possibility for this study to accord the proper amount of time necessary, the results would be reduced to preliminary. But, even in this case, the outcome seems to be overly enthusiastic, as the enhancements add a performance boost event from an early training stage.

Contrary to expectations set in the beginning, the Spatial-Attention enhancement seems to have a bigger effect on the performance compared to the entropy model (the expectations were that the spatial attention should not boost the results that much, but rather to feed the proper information to the entropy coder, so that the later one will offer the true boost in performance). The reasoning behind it could be described as the training of the new entropy module requires more time in order to learn how to really take advantage of the spatial data offered by the feature extraction. But even with such a small training time there still seems to be an increase in performance, which leads to the belief of increased effectiveness after a proper training session.

In Table 2, it can be seen in detail the overall performance of the models. The PSNR, MSSIM, MES, and BPP values are obtained by averaging all the values coming from the total test dataset. It is important to pay attention to these values as it shows constant improvements, which further proves the effectiveness of the propped method, and helps. If there is an improvement in the model performance for the measured metrics it is pointed with a green arrow next to the obtained value.

Table. 2 Overall performance of the models

Model	PSNR Y	PSNR U	PSNR V	PSNR YUV	MS-SSIM RGB	MSE RGB	PSNR RGB	BPP
FVC	37.34	42.73	43.87	39.33	0.955	227.72	35.89	0.0467
FVC-SA	37.34	42.80 ↑	43.98 ↑	39.36 ↑	0.955	226.78 ↑	35.91 ↑	0.0466 ↑
FVC-SA-E	37.36 ↑	42.88 ↑	43.90 ↑	39.37 ↑	0.955	226.24 ↑	35.93 ↑	0.0464 ↑

5. Conclusion

The final project we do this time is group homework, which requires every member to corroborate. However, we encountered a lot of difficulties during this process, such as communication problems. Because there are not only Taiwanese but also foreign students in our group, our communication at the beginning was not that good. We often misunderstand the meaning of each other. In addition, our baseline should not be image compression at the beginning, which takes us so many times to find a new one. Fortunately, everyone communicates activity in the later stage. We asked each other if we had any questions, which let our work gradually get on track. Finally, we completed the final project successfully.

6. REFERENCES

[1] Eirikur Agustsson, David Minnen, Nick Johnston, Johannes, Sung Jin Hwang, and George Toderici. Scale-space flow for end-to-end optimized video compression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 8503–8512, 2020.

[2] Zhihao Hu, Zhenghao Chen, Dong Xu, Guo Lu, Wanli Ouyang, and Shuhang Gu. Improving deep video compression by resolution-adaptive flow coding. Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[3] Zhihao Hu, Guo Lu, and Dong Xu. FVC: A New Framework towards Deep Video Compression in Feature Space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 1502–1511. 2021.

[4] Jiahao Li, Bin Li, Yan Lu. Hybrid Spatial-Temporal Entropy Modelling for Neural Video Compression. In Proceedings of the 30th ACM International Conference on Multimedia. page1503–1511,2022.

[5] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional Block Attention Module. arXiv preprint arXiv:1807.06521,2018.

[6] David Minnen, Johannes Ballé, and George D Toderici. 2018. Joint autoregressive and hierarchical priors for learned image compression. Advances in neural information processing systems 31,2018.

7. Work distribution chart

Team member	Work content
Manu	(1) Project baseline program running
	(2) Baseline model optimization job
	(3) Project report writing and typesetting
	(4) Thoughts and Discussion on Program Project Optimization
柯佐勳	(1) Thoughts and Discussion on Program Project Optimization
	(2) Improved optimizer model (Entropy model)
	(3) Project report writing and typesetting
林佑珊	(1) Thoughts and Discussion on Program Project Optimization
	(2) Dataset establishment and initial operation
	(3) Project report writing and discussion