A Comprehensive Analysis of Video Frame Interpolation Artifacts, Evaluation Metrics and Potential Enhancements

Debanjan Mondal
debananjanmond@umass.edu

Suraj Pathak
sspathak@umass.edu

Avinesh Krishnan
avineshkrish@umass.edu

February 16, 2024

Abstract

Video frame interpolation is a challenging computer vision problem that aims to artificially increase the frame rate of videos. VFI has been explored extensively in the past. Our goal is to synthesize intermediate frames between two adjacent frames of a video to produce high quality videos with smooth view transition. We intend to use VFI model's excellent abilities to learn mapping between two consecutive frames and the intermediate frame, and reduce the issues brought on by occlusions and different lighting conditions. VFI faces numerous challenges because the interpolated frames are frequently overly smoothed or blurred, and have artifacts. We evaluate three state of the art models when presented with challenging scenarios like large motion, heavy occlusion and presence of high frequency details in the image. We also propose an alternative metric that does a more holistic evaluation of the interpolated frames and conduct experiments to confirm whether the metric performs well.

1 Introduction

Creating intermediary frames between two consecutive video frames is the goal of video frame interpolation. Applications including slow-motion creation, video compression, and unique view synthesis are achieved by VFI. Real-time VFI algorithms operating on high-resolution videos also have a wide range of potential applications, including lowering the bandwidth requirements for live video streaming, offering video editing services to users with constrained computing resources, and video frame rate adaptation on the display devices.

The complicated, huge non-linear motions and lighting changes in real-world videos make VFI difficult. Motion-based algorithms have undergone the most active development in light of the most current developments in optical flow estimation. By warping two succeeding frames in either direction, they may predict an intermediate frame using optical flows. Deep convolutional neural networks have achieved considerable success in video frame interpolation, although there are still certain limitations: The resulting frames contain artifacts like ghost effect and blur. These artifacts are brought on by weaker picture fusion and imprecise motion estimates. Deep neural networks feature complicated computations and vast model sizes. As a result, deploying the models to hardware with limited storage and processing power is challenging.

2 Related Works

The mainstream VFI methods can be classified into optical flow-based methods and kernel regression methods. Both types of methods have their advantages and disadvantages. For example, kernel-based methods are good at handling motion blur by convolving over local patches. We looked at one such paper named IFRNet[4]. It combines motion estimation and frame synthesis into a single convolution step. It estimates a pair of 2D convolution kernels and uses them to convolve with previous frame and the next frame to compute the color of the output pixel. It further reduces the computational complexity by producing two 1-D kernels to approximate a 2 D kernels. The model uses a fully convolutional neural network.

However, kernel based methods are typically computationally expensive and short of dealing with occlusion. Also these can't capture long range motions due to their limited receptive field. Optical flow based methods utilize optical flow to build pixel wise correspondences, thereafter, they warp the given neighboring frames to the target frame. Recently deep learning based approaches are used to estimate optical flows. Another paper [2] uses encoder-decoder based architecture in this task. This method extracts pyramid features from images using encoders and then gradually refine intermediate flow fields through multiple decoders by backward warping pyramid features.

Paper [11] further introduces the concept of texture consistency loss instead of a pixel wise losses with the ground truth images, arguing that given previous and next frames, the solution of the intermediate frame prediction may not be unique. It used a cross-scale pyramid alignment across different scales of features using deformable convolution networks to maintain temporal consistency. Finally it used an attention based model to fuse features and generate intermediate frames.

One approach to image synthesis that has recently garnered a lot of attention is the concept of diffusion based models. In a recent paper [1], OpenAI scientists Dharival and Nichol show that diffusion based models beat state of the art generative models and produce a much higher quality output. Diffusion based models work by progressively adding noise to the data to the point where it is almost pure noise. A model is then trained to iteratively denoise the data to finally obtain a clean sample. During the denoising process, high frequency details are synthesized in the image. In the context of video frame interpolation, the quality of the synthesized image largely determines the quality of the final interpolated video produced.

Few researchers have attempted to synthesize clear, high frame rate outcomes from blurry, low frame rate inputs, an issue common to video enhancement. In order to eliminate blur and increase frame rate at the same time, this paper [8] proposed an interpolation technique for blurry video frames. Modern frame interpolation techniques typically perform frame warping to synthesize pixels using reference frames after initially estimating an object's motion. However, the motion estimation could not be precise if the original reference frames are compromised by motion blur. Frame deblurring followed by frame interpolation is a direct response, although the interpolation quality is not as good as it may be. Since the interpolated frames include blurry input textures, doing frame interpolation and then frame deblurring also reduces overall quality. The research paper [8] suggests two modules made up of various backbone networks that together make up the unified Blurry video Interpolation (BIN) approach. The spatial consistency between the input frames and the regenerated frames of the modules is enforced by the model using pixel reconstruction and cycle consistency losses. The module processes temporal information using ConvLSTM units, which enables it to synthesize images with consistency and recover fine details. Thus, their model outperforms state-of-the-art techniques and completely utilizes space-time information.

3 Methodology

Since Video Frame Interpolation is a very challenging task, we started by looking at some state-of-the-art models. We picked 3 models - IFRNet[2], FilmNet[7] and ABME[5]. We give a brief overview of the models here

- **IFRNet:**[2] This is an optical flow based model which performs an extraction phase so as to retrieve a pyramid of features from each frame. It then gradually refine intermediate flow fields through multiple decoders by backward warping pyramid features. Apart from an $L1$ reconstruction loss, it calculates loss of the predicted flow in each decoder layer with the flow output of a off-the-shelf teacher flow network, that helps to align multi-scale pyramid features explicitly. It also calculated geometric consistency loss similar to local binary patterns to retain the local geometric shape.

- **FilmNet:**[7] This is another optical flow based model that uses a scale-agnostic motion estimator in order to interpolate frames for cases with large motion. Additionally, the model uses a ”Gram matrix” loss that measures the correlation difference between features. This model uses VGG-19 features in order to better estimate loss.

- **ABME:**[5] In ABME, we predict symmetric bilateral motion fields and refine them by loosening the linear motion constraint. Specifically, we interpolate a temporary intermediate frame using the symmetric fields and then estimate asymmetric bilateral motion fields from the anchor frame to the two input frames. In the frame synthesis, the input frames are warped using the bilateral motion fields and aggregated using two subnetworks, FilterNet and RefineNet. FilterNet generates dynamic filters to exploit local information while RefineNet reconstructs a residual frame using global information. ABME handles occluded regions effectively and provides excellent interpolation results.

3.1 Dataset

We picked up a large-scale dataset called OVIS[6] for occluded video instance segmentation. This video dataset was originally designed for video instance segmentation. The most distinctive property of the OVIS dataset is that a large portion of objects is under various types of severe occlusions caused by different factors. The average video duration and the average instance duration of OVIS are 12.77s and 10.05s respectively which are quite long compared to VFI benchmark datasets. The scenes are often very crowded. On average, there are 5.80 instances per video and 4.72 objects per frame. This makes it an excellent dataset for finding failure cases for video frame interpolation.

3.2 Artifact Detection

Since the current benchmark datasets in VFI don't contain crowded scenes and severe occlusions, we applied the models mentioned earlier in the OVIS dataset and did a frame by frame analysis to detect artifacts that are present in the intermediate frames. We present these artifacts and comparative analysis of the models in 4.

3.3 Proposed Loss and Evaluation Metric

Finally after idetifying the artifacts in the synthesized images, we propose an Object Detection based loss and evaluation metric that can be used to train and evaluate Video Frame Interpolation models. Due to the size of the VFI datasets and lack of computing sources, we couldn't train a model with this loss. The details are described in 5.

4 Experiments

4.1 Temporal Pixel Correlation

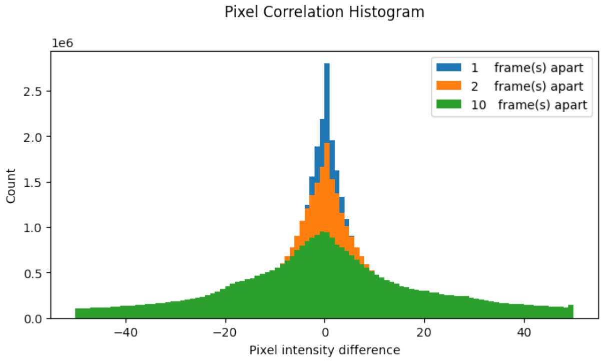


Figure 1: Temporal correlation of pixel intensity as a histogram

We conducted an experiment to visualize the temporal correlation of pixel intensities. We randomly selected 8 videos from the OVIS dataset and then randomly selected 5 frames from each video. We then computed the difference in the pixel intensities of each of the 5 frames with that of the next frame, two frames ahead, and 10 frames ahead. The images were converted to grayscale to obtain the pixel intensities. Finally, we grouped the pixel intensity difference values by the distance between the frames before plotting a histogram. It is evident from the histogram shown in [1] that as we increase the temporal gap between two frames in a video, the farther the frames are, the lower the correlation is between the two frames. The peak at 0 in the histogram indicates that the pixel values do not vary as much for a frame distance of 1. However, in the case of a frame distance of 10, the histogram is a lot more flatter. This result affirms our intuition that frames that are temporally closer to one another have similar pixel distributions.

4.2 Motion

Fast moving objects present a challenging task for video frame interpolation. Often times, the complete object may not be present in both the next or previous frames. Instead, only a part of the object is visible in one of the frames with the complete object in the other. In such cases, a VFI model has to be able to determine how much of the object should be visible in the interpolated frame and accordingly synthesize the middle frame. We manually selected a very challenging video with large motion and generated interpolated frames using all three models. The pair of frames [2a 2b] contains large motion of the black car as it passes in front of the camera The interpolated frames from each model contain different types of artifacts. Looking carefully at the image generated by FILMNet, we can roughly see some structure that could resemble a car. The headlight and fog light is somewhat visible although it does not have the same shape. The interpolated frames from IFRNet and ABME models fail to clearly draw the car and instead we see a blob that matches the color of the car.

4.3 Patterns

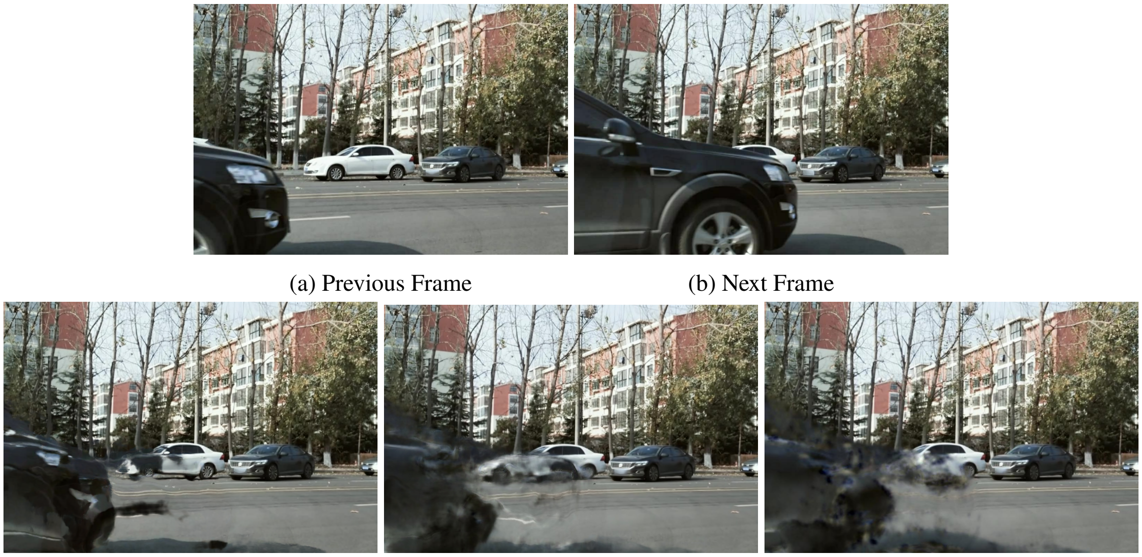
Pattern artifacts are evident when there is a moving object behind a particular structure with a pattern. For example, a person moving behind a fence or a gate. In such cases, the VFI model has to account for the fact that the structure of the fence needs to stay constant while the object moving behind it may not be completely visible. Thus, patterns present a challenging task for VFI models. We chose a video from the OVIS dataset containing birds moving around in a cage. The pair of frames [3a 3b] contains a bird on the top right moving towards the left side of the cage. The interpolated frames generated by each of the three models have distortions. In case of the FILMNet model we see that the model does not preserve the structure of the bird cage. Looking closely at the image in 3c we can see that the cage is disconnected in the portion where the bird is moving. This, however, is not a problem in the case of IFRNet and ABME. Both models perform well in terms of keeping the structure of the cage intact.

4.4 Texture

Textures are repeating high frequency details in an image. For example, bird feathers, bubbles in a fish tank, etc. Textures present an interesting challenge for VFI as it is very easy to apply heavy denoising on the image and achieve a low MSE. This is a well known drawback of the MSE loss and PSNR evaluation metric. In the interpolated frames shown in [4c 4d 4e], IFRNet and ABME seem to be applying a strong smoothing effect thereby erasing the bubbles from the image. FILMNet is the only model that generates images where the air bubbles are clearly visible. However, we noticed a temporal artifact when playing the video with the interpolated frames. While each image individually contains air bubbles rendered in good quality, the motion of the air bubbles is not natural and the bubbles appear to be moving about in the same spatial location. Furthermore, in [5c 5d 5e], the fluffy appearance of the goose's feathers in the original frames [5a 5b] is not present in the interpolated frames generated by IFRNet and ABME. The models seems to be applying heavy smoothing on the synthesized images. FILMNet, on the other hand, performs better in terms of preserving high frequency information as the feathers look more natural.

4.5 Occlusions

Occlusions are yet another challenging problem for computer vision because they cause objects to partially or completely be blocked by another object. We noticed a variety of artifacts for frames with occlusions. We selected a very challenging video for this task. The input frames in [5a 5b] have a goose that is lowering its neck. While doing so, its body stays in the same location but its neck moves down. As a result, we have an object that is effectively changing shape while revealing another object in the background. The artifacts for occluded objects were different for IFRNet and ABME when compare to those in FILMNet. As we noted in examples discussed previously, IFRNet and ABME apply a high degree of smoothing to the image. Although they seem to be accurately predicting how much of the neck should be visible in the interpolated frame, there is a clear loss of high frequency details of the feathers. The frame generated by FILMNet preserves high frequency information but deforms the goose's head. In another example where two birds were in front of one



(a) Previous Frame

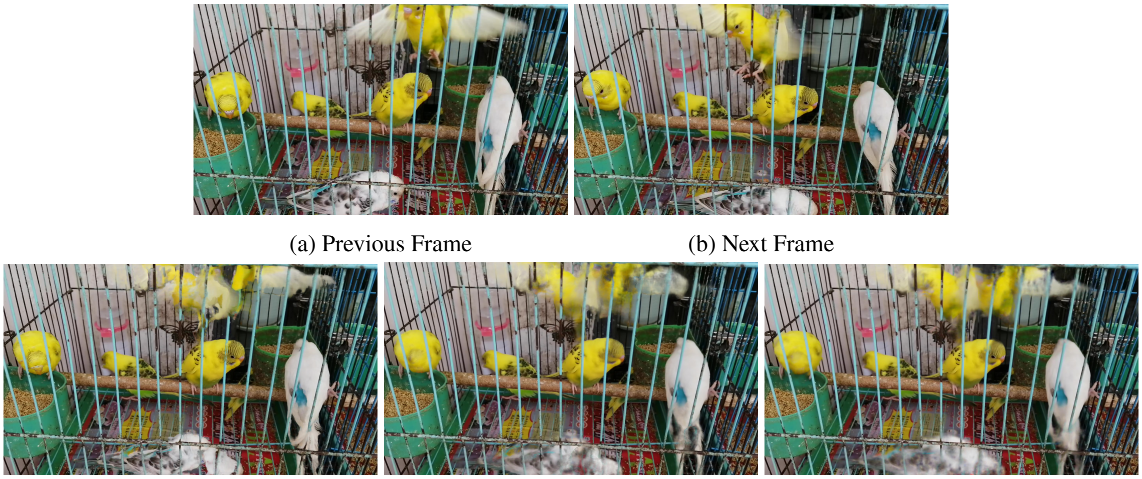
(b) Next Frame

(c) FILMNet

(d) IFRNet

(e) ABME

Figure 2: Texture artifacts



(a) Previous Frame

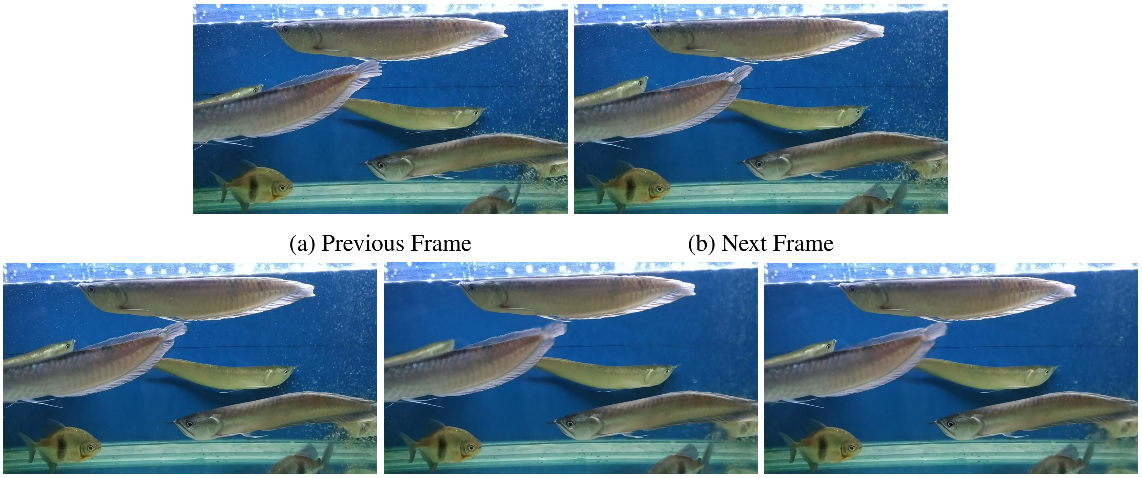
(b) Next Frame

(c) FILMNet

(d) IFRNet

(e) ABME

Figure 3: Pattern artifacts



(a) Previous Frame

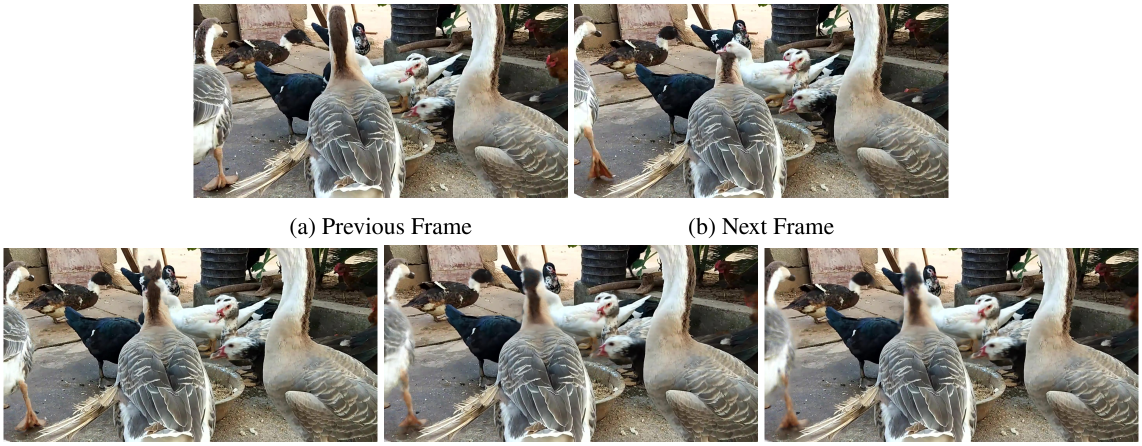
(b) Next Frame

(c) FILMNet

(d) IFRNet

(e) ABME

Figure 4: Loss of high frequency information like Bubbles



(a) Previous Frame

(b) Next Frame

(c) FILMNet

(d) IFRNet

(e) ABME

Figure 5: Artifacts in occlusion

another, the interpolated frame created by FILMNet had an illusion that made it seem as if the birds were swapping heads. For the same example, the frames generated by IFRNet and ABME did not show this type of artifact. However, there was a clear loss of high frequency information as the bird feathers were heavily smoothened.

5 Proposed Approach

5.1 Evaluating MSE and PSNR metric

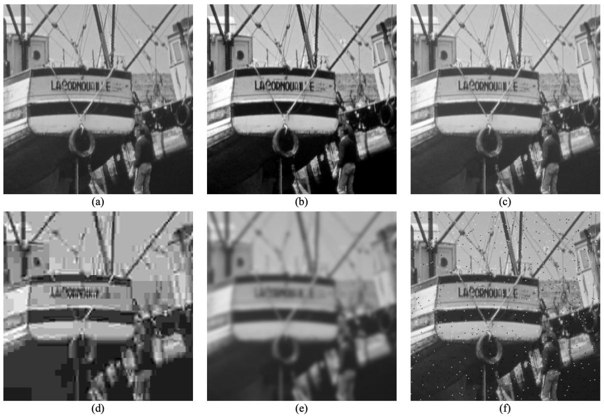


Figure 6: Visually dissimilar images with similar MSE scores

MSE and PSNR are two of the most popular metrics used to determine the quality of images produced by computer vision models. However, it is a well known fact that these metrics do not accurately reflect the perceived quality of an image [10]. It is evident from the figure [6] that the images differ vastly in terms of the quality. However, the authors of the paper declare that all of the images have the same MSE score [10].This result suggests a need for an improved evaluation metric for comparing the quality of images in addition to a loss function that can reflect image quality more accurately as perceived by the human eye.

We can clearly PSNR has certain limitations in capturing the quality of generated frames. PSNR is inversely proportional to the noise in the image. In order to outperform other models in this metric, models like IFRNet produces smoothed image in which the high frequency information about the image is lost. Also in some cases, as shown in the last section, important semantic information is also lost. So even though a model is considered best in the PSNR metric, it might not be the best when subject to human evaluation.

Currently we have very accurate object detectors. Since object detectors capture semantic information of the image, we can evaluate the image quality by the performance of a pre-trained object detector on that image. In the following sections we proposed a objection detection based loss which can be used while training and an evaluation metric which can be used to evaluate the performance of a VFI model.

5.2 Object detection based loss

The motivation behind designing this loss function is that the synthesized middle frame should have similar objects as the previous and next frames. We picked up pre trained Yolov7[9] object detector pre-trained on the COCO dataset[3].

We first run the object detector on the previous and next frames and extract the bounding boxes and corresponding most probable classes. We used a confidence threshold of 0.25 to disregard bounding boxes with low confidence. Then we derive correspondence between the bounding boxes by nearest neighbor search using the euclidean distance between the 4D coordinates vectors of the bounding boxes. To reduce the computational cost, we do the nearest neighbor search in class by class basis for both images. Assuming that motion in small timeframe is linear, we can approximate the bounding boxes in the middle frame by taking simple average of the bounding box coordinates of the previous frame and next frame. Now we have a pseudo ground truth target.

We then run the object detector on the synthesized frame and extract the bounding boxes and corresponding class probabilities. Here also we used a confidence threshold of 0.25 such that most probable class in each bounding box should have at least 0.25 probability, otherwise we discard that bounding box. Again we find correspondences with the pseudo ground truth and predicted bounding boxes, this time all vs all, not considering the class information. We then calculate the bounding box loss which is the just the mean squared error between the pseudo ground truth and predicted bounding box coordinates. We also calculate the Class loss which is the Cross entropy loss between the predicted class probabilities and pseudo ground truth class label. Our final loss is the weighted sum of the two losses.

$$Loss = \frac{1}{N}\left(\sum_{i=1}^{4}(PsBB(i) - PrBB(i))^2 \right.$$
$$\left. + k * CrossEntropy(PsCl, PrCl)\right)$$

We set $k = 0.5$ for the following results. In 7 we present the losses on the car images produced by different models. We can see that even though all the outputs are blurry, the output produced by FilmNet retains the most semantic information about the car whereas IFRNet and FilmNet produce a blurry blob. The proposed loss metric is in line with the visual quality, the loss is lowest for the FilmNet output, and is similar for the ABME and IFRNet outputs.

In 8 we present the losses on the goose images produced by different models. We can see the IFRNet output preserve the most semantic information about the geese, for example, the beak is quite clear for the central duck in this output, even though the output is blurry. For the other outputs the beak is not clear. The proposed loss is in line with the expectation.

5.3 Object Detection Based Metric

Here we also propose an alternate metric to PSNR for evaluating Video Frame Interpolation Models. We can pick up a pre-trained object detector, in this case we picked up a Yolov7[9] object detector pretrained on the COCO dataset[3]. While evaluating we have access to the ground truth intermediate frames. We ran the object detector on these ground truth images and extract the bounding boxes and the corresponding most probable class for each class. We used a confidence threshold of 0.25 to disregard bounding boxes with low confidence.

Next we run the object detector on the predicted images. Having the bounding boxes and classes as our target, we can calculate the Mean Average Precision(mAP) over all classes. mAP is a widely used metric to evaluate object detection models. For a well established object detector, we argue that this metric captures the quality and semantic information of the synthesized intermediate frames.

We calculated this metric for IFRNet and FIMNet on Vimeo90K and SNU-FILM Hard datasets. The results are presented in Table 1. Here mAP@0.5 is the mean average precision calculated by taking 0.5 as the IOU threshold. mAP@.5:.95 represents taking IOU threshold = 0.5, 0.55, 0.60, ...0.95 with steps of 0.05, calculating mAP each threshold and finally averaging them. We can see that although IFRNet performs better in terms of PSNR metric, FilmNet performs better in terms of mAP. This metric gives us idea about the synthesized object quality. Certain models like IFRNet which heavily smoothes the image to get higher PSNR, might not perform well in this metric.

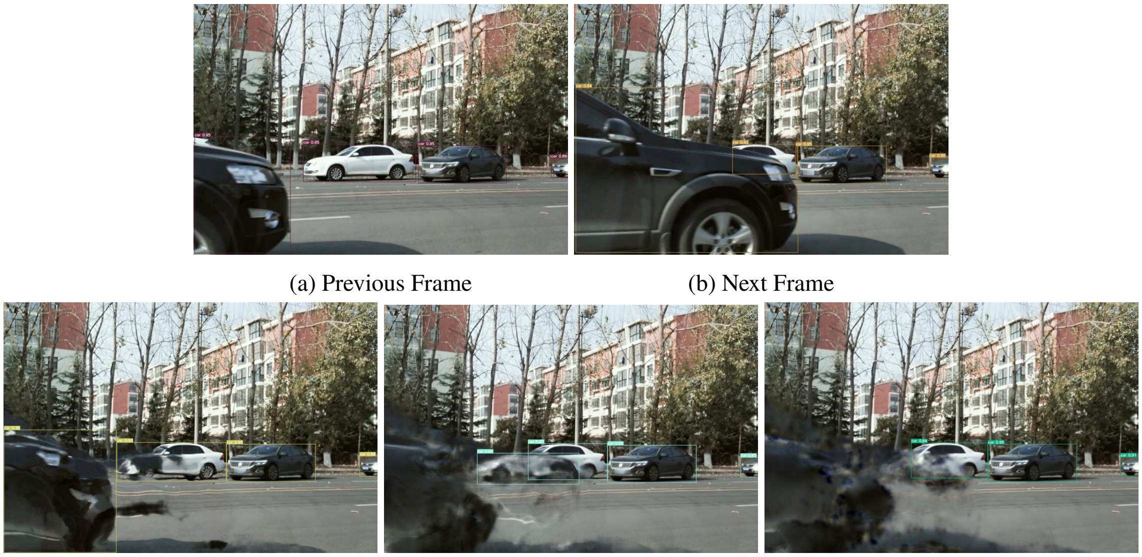
6 Future Work

6.1 End to End Training

An existing model can be trained with the proposed Object Detection Based loss and we can compare the trained model with the existing state-of-the-art models with PSNR metric as well as the proposed Object Detection based metric.

6.2 VFI Based Video Compression

A popular use case for video frame interpolation is data compression. With a reliable VFI model, one could theoretically cut down storage consumption by 50% by dropping every alternate frame and generating the missing frames on demand. However, this becomes challenging as dropping frames makes it more difficult to estimate optical flow and consequently results in poorer video frame interpolation results. Instead of dropping alternate frames entirely, we could drop a certain % of the pixels from each frame under the constraint of minimizing error in optical flow measurement. We hypothesize that this strategy would improve the quality of video frame interpolation as the model will have more data about the frame it needs to interpolate. The pixels that are not dropped can be used by the model as a heuristic for optical flow measurement and also improve the quality of synthesized high frequency details in an image. Furthermore, we could apply this type of pixel dropping for each of RGB channels separately. Finally, we could use a temporal demosaicing strategy that uses multiple frames to synthesize the missing pixels for each channel.



(a) Previous Frame

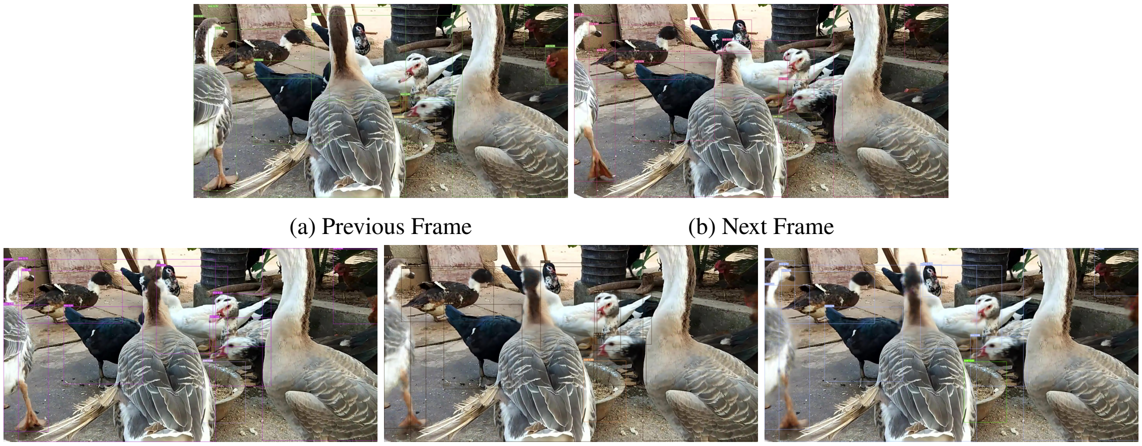
(b) Next Frame

(c) FILMNet Output: Loss = 0.0171

(d) IFRNet Output: Loss = 0.0250

(e) ABME Output: Loss = 0.0255

Figure 7: Object Detection Losses for Car



(a) Previous Frame

(b) Next Frame

(c) FILMNet Output: Loss = 0.0967

(d) IFRNet Output: Loss = 0.0919

(e) ABME Output: Loss = 0.0951

Figure 8: Object Detection Losses for Goose

Model	Dataset	PSNR	mAP@0.5(Yolov7)	mAP@.5:.95(Yolov7)
IFRNet	Vimeo90K	35.80	0.858	0.755
IFRNet	SNU-FILM Hard	30.41	0.705	0.618
FilmNet	Vimeo90K	35.76	0.868	0.767
FilmNet	SNU-FILM Hard	30.20	0.723	0.629

Table 1: Comparison between PSNR and mAP for different models and datasets

6.3 Bounding Box Texture Quality Analysis

In our bounding box based metric, we do not measure the quality of the synthesized object's texture. As a result, any loss of high frequency information such as textures, edges and other fine details is ignored. For example, say, an interpolated image of a car causes it to lose its vinyls, or changes the color of the car. Our bounding box regression loss approach would not accurately reflect this as loss of quality as long as One of the next steps for our project could be to use a texture comparison metric within the bounding boxes. Factoring in texture information into the final evaluation score would allow us to better evaluate the ability of the model to preserve high frequency information such as bird feathers, bubbles in a fish tank, edges or other stylistic features of cars, etc.

6.4 Optical Flow Aided Object Detection and Recognition

One of the challenges we faced while building the object detection based evaluation metric was that some images had too many objects occluding one another. As a result, the object detector we used was not able to accurately detect all objects. To improve on this, we could use an alternative object detector that can factor in additional details from nearby frames. Since we are working on a problem in the domain of videos, we can assume we have access to multiple frames unlike YOLOv7 which only considers object visible in a single frame. An evaluation metric that uses optical flow could improve object detection and, consequently, give measure the quality of the interpolated frame .

References

[1] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021.

[2] Lingtong Kong, Boyuan Jiang, Donghao Luo, Wenqing Chu, Xiaoming Huang, Ying Tai, Chengjie Wang, and Jie Yang. Ifrnet: Intermediate feature refine network for efficient frame interpolation, 2022.

[3] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014.

[4] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution, 2017.

[5] Junheum Park, Chul Lee, and Chang-Su Kim. Asymmetric bilateral motion estimation for video frame interpolation, 2021.

[6] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip Torr, and Song Bai. Occluded video instance segmentation: A benchmark. *International Journal of Computer Vision*, 2022.

[7] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *European Conference on Computer Vision (ECCV)*, 2022.

[8] Wang Shen, Wenbo Bao, Guangtao Zhai, Li Chen, Xiongkuo Min, and Zhiyong Gao. Blurry video frame interpolation, 2020.

[9] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*, 2022.

[10] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[11] Kun Zhou, Wenbo Li, Xiaoguang Han, and Jiangbo Lu. Exploring motion ambiguity and alignment for high-quality video frame interpolation, 2022.