Fast Frame Rate Up-conversion Using Video Decomposition

Nikhil Singh and Saumik Bhattacharya

***Abstract*—Video is one of the most popular medias in the world. However, video standards that are followed by different broadcasting companies and devices differ in several parameters. This results in compatibility issues in different hardware while handling a particular video type. One of such major, yet important parameter is frame rate of a video. Though it is easy to reduce the frame rate of a video by dropping frames at particular interval, frame rate up-conversion is a non-trivial yet important problem in video communication. In this paper, we apply video decomposition algorithm to extract the moving regions in a video and interpolate the background and the sparse information separately for a fast up-conversion. We test our algorithm for different video contents and establish that the proposed algorithm performs faster than the existing up-conversion method without producing any visual distortion.**

***Index Terms*—Frame rate up-conversion, Video Decomposition, Video Interpolation**

I. Introduction

.

IN last years, internet and multimedia technology is pervading. This is increasing the urge for video streaming and video conferences alongside limited availability of bandwidth that creates constraints on the use of internet. In order to adapt with internet constraints on available bandwidth, a new codecs is required using which video can stream at available transmission rate but it may result in loss of both spatial and temporal quality. Several methods have been proposed in spatial domain techniques with constrained bandwidth but it produces blurring effect and blocking artifacts and hence, those methods were failed to preserve detailed quality of the video. Then, the idea reached to temporal domain from spatial domain. With the limitation of bandwidth, transmission of entire video frames is not possible from encoder that forced to use sub-sampling techniques to reduce the number of transmitted frames.

Human eyes have tendency to see in integrated manner. Skipped frame in video sequence resulted in jitter effect hence, it creates inconvenience to human perception. Consequently, it motivates towards algorithm that take sub-sampled video as input at encoder and reconstruct the skipped frames at decoder side. Hence, in order to get best possible perceptual quality video, a method called frame interpolation has been proposed.

In this pervasive world, expansion has been seen in distinct types of multimedia devices, each may have distinct features like resolution, frame rate formats, etc. In broadcasting and displaying technology, efficient video conversion algorithms are required to handle device compatibility and to perform better in constrained environment like power consumption and execution time. Algorithms which satisfy required criterion effectively, commonly known as frame rate up-conversion (FRUC) techniques. In LCD, motion blur and drag effect in video arise for fast moving images due to slow response and physical limitation of LCD that can be reduced by using FRUC techniques on the decoder side of LCD [1]- [2]. FRUC can also be applied in scalable video coding, slow motion video, and so on.

For real life utilization, several FRUC techniques have been proposed under distinct complexity [3]- [13]. Classically, frame duplication/frame averaging and motion estimation/motion compensation (ME/MC) are two kinds of techniques which are used as FRUC algorithm. In frame duplication/frame averaging algorithm, intermediate or skipped frames are obtained by computing the mean between previous and next frames and sometimes by exploiting various linear interpolation methods. This method was popular due to its simple computation, but for fast moving objects, it produces blur and jitter effect as movement of scene is not taken into consideration. Motion compensated frame rate up conversion (MC-FRUC) is a widely practiced algorithm for estimating intermdiate frames under (ME/MC) category [10]. MC-FRUC techniques can be used in two steps to exploit the correlation between temporal and spatial domain of frames, which is used to obtain intermediate or skipped frame. First step is motion estimation (ME), using which motion of an object can be represented as motion vectors (MVs) and second is motion compensated frame interpolation (MCI) which uses generated MVs to interpolate the intermediate frames.

Block-matching algorithm (BMA) is a widely accepted technique for motion estimation (ME) due to its less complex implemenation. In BMA, image get segmented in blocks and then movement of those blocks are detected. Unilateral ME and Bilateral ME are the two primary kinds of ME which are being used in BMA. Using unilateral ME, many FRUC algorithms have been proposed in literature [5]–[7], [14], [15]. To obtain MVs for frame interpolation, unilateral ME pass in one direction and search for the most similar block in reference image, limited by the range of search window [37], which resulted in overlaps and holes. Many algorithms have been proposed to reduce hole regions [8], [16] that consider spatial relation in interpolated frames but results of their interpolations were degraded (not smooth) especially, when large hole regions appeared or hole regions existed between objects. Then, to primarily solve the problem of holes and overlapped regions, a new FRUC techniques using bilateral ME (BME) was proposed [9]. Two-way unidirectional FRUC was proposed in [17], [18] in order to handle hole regions and occlusion caused by incorrectly estimated motion vectors, but these approaches still did not completely remove them.

Several other approach have been proposed in literature to improve the accuracy of BME hence to overcome the problem of holes and overlapped regions. Extended bilateral motion estimation (EBME) [11] is another such approach in which extra Mvs are calculated by overlapping extra reference block with each adjacent original reference block by factor of half and hence to get more true MVs with greater probability as compared to approach conventionally used in BME techniques. In Dual ME technique [18], the motion vector field (MVF) of interpolated frame is tuned by using the unidirectional and bidirectional block matching ratios in previous and next subsequent frame. In direction-select ME (DS-ME) technique [18], motion vectors (MVs) are calculated for both forward and backward directions by independently estimating motion in both directions. Based on the value of sum of bilateral absolute differences (SBAD) for each MV more reliable MV is selected for frame interpolation.

Another approach called Multi-Channel Mixed-Pattern (MCMP) [21] was proposed which use variant of 3DRS algorithm to estimate initial MVFs. During tuning process of spatio temporal MVs, smoothing constraint applied to obtain more smoother value of MV. Then highly fault-tolerant motion vector smoothing (HFT-MVS) technique is used to overcome the problem of outliers in MVF. Finally, by utilizing MVFs dual-weighted overlapped block motion compensation (DW-OBMC) techniques is used, to reduce the problem of blurring effect and occlusions. Although using above-mentioned algorithm more accurate MVF can be obtained to increase the performance but it also increases expense of computational complexity.

Due to constraints on resources and execution time, block-based ME algorithms have been widely practiced in video compression and processing techniques. Block-based ME is used to estimate block motion vector between subsequent frames which is then used to predict a coded block between adjacent frames, commonly know as inter frame prediction. This differencing process of calculating difference between coded block and its predicted motion vector is defined as Motion compensation (MC) . Resultant of this process arises block prediction error, also known as MC-residual [19]. To exploit residual spatial redundancy, MC-residual is further coded using transform coding and feed to the decoder. This results in blocking artifacts and false edges. Performance of inter frame coding greatly affected by MC- residual hence, the overall performance of compression.

In this paper, a new frame rate up-conversion scheme is proposed to overcome the problem of the estimation of ME/MCI, selection of adaptive window size or filters w hich i s t o be used for comparing different blocks positions in corresponding frames and computational cost for various types of ME. In the proposed scheme, previous, current and existing frames are being utilized to implement 1-D cubic spline interpolation algorithm which is used to construct the missing frames to be interpolated. The proposed scheme is suitable for the parallel processing and pixels are independent of neighboring pixels while constructing or estimating the new frames. Consequently it helps to avoid problem of holes and overlapped regions and hence it avoids any computationally extensive method like the method based on MVs and ME. Also in earlier methods, additional regularization is required as smoothness constraint on motion field while estimating MVs but spline introduces smoothness constraint implicitly on motion field which helps in avoiding extensive application of regularization.

Remaining of this paper is organized as follows. In Sec. II, we discuss the proposed algorithm. In Sec. III, we demonstrate the performance of the proposed algorithm for different input videos. Finally, we conclude the paper in Sec. IV discussing the advantages, limitations and future scopes of the proposed method.

II. Proposed Algorithm

The proposed frame up-conversion algorithm has two parts- first we decompose an input video into background video and feature video. Background indicates the information which is static or slow varying in the video shot. Then, we use spline interpolation on the feature part of the video to upsample a video volume.

A. Video Decomposition

Let us assume that for a given video $\mathbf{V}$ with N number of frames and frame resolution $P \times Q$, a particular pixel $\mathbf{p} = (x, y)$ belongs to the background of the video. The intensity at pixel location $\mathbf{p}$ for all frames should not vary between two neighbouring frames if $\mathbf{p}$ is located at the background. In other words, if we calculate a vector $\mathbf{x_p}$ such that

$$\mathbf{v_p} = [v_p^1 - v_p^2, v_p^2 - v_p^3, \ldots, v_p^{N-1} - v_p^N]^t \tag{1}$$

the vector $\mathbf{x_p}$ will be a sparse vector as $\mathbf{p}$ is a background pixel, where v_p^i is the intensity at location $\mathbf{p}$ in the ith frame of $\mathbf{V}$. We can rearrange eq. 1, as $\mathbf{x_p} = \mathbf{M}\mathbf{v_p}$, where $\mathbf{M}$ is the variation matrix, defined as-

$$\mathbf{M} = \begin{bmatrix} 1 & -1 & 0 & \ldots & 0 \\ 0 & 1 & -1 & \ldots & 0 \\ & & & \ddots & \\ 0 & 0 & \ldots & 1 & -1 \end{bmatrix}_{(N-1)\times N}$$

Generalizing this idea, if we construct a vector $\mathbf{z_p}$ of dimension $N \times 1$, for any pixel location $\mathbf{p} \in P \times Q$, such that z_p^i, the i^{th} element of the vector which represents the intensity at pixel location $\mathbf{p}$ in i^{th} frame. Our objective is to estimate $\mathbf{v_p}$ from $\mathbf{z_p}$ such that $\mathbf{M}\mathbf{v_p}$ is a sparse vector. This estimation gives us the background intensity at pixel location $\mathbf{p}$ for all N frames. To estimate $\mathbf{v_p}$, we define the problem as

$$\begin{aligned} \underset{\mathbf{v_p}}{\text{minimize}} \quad & \{\|\mathbf{z_p} - \mathbf{v_p}\|_2^2 + \lambda \|\mathbf{M}\mathbf{v_p}\|_0\} \\ \text{subject to} \quad & \lambda \geq 0 \end{aligned} \tag{2}$$

where $\|.\|_p$ denotes l_p norm of a vector. The first term of the expression is the data fidelity term, the second term regularize

the smoothness of the estimated vector $\mathbf{v_p}$ and λ determines the level of smoothness.
A well-known method to solve the non-convex problem, mentioned in eq. 2 is non-convex, is to replace the l_0 norm with l_1 norm [28] to form a convex problem. We modify the optimization problem of eq. 2 accordingly as

$$\begin{aligned} &\underset{\mathbf{v_p}}{\text{minimize}} \quad \{\|\mathbf{z_p}-\mathbf{v_p}\|_2^2+\lambda\,\|\mathbf{M}\mathbf{v_p}\|_1\} \\ &\text{subject to} \quad \lambda \geq 0 \end{aligned} \tag{3}$$

To solve for the minimization problem given by eq. 3, first, we define a cost function $C(\mathbf{v_p})$ as

$$C(\mathbf{v_p}) \;=\; \min_{\mathbf{v_p}} \{\|\mathbf{z_p}-\mathbf{v_p}\|_2^2+\lambda\,\|\mathbf{M}\mathbf{v_p}\|_1\} \tag{4}$$

It can be observed that l_1 norm is not differentiable. Thus the cost $C(\mathbf{v_p})$ can not be minimized analytically. However, it can be shown that for any variable u,

$$|u| = \max_{-1\leq l\leq 1} lu \tag{5}$$

where $|.|$ represents the absolute value of a variable. Using the equality, we can write that

$$\|\mathbf{y}\|_1 = \max_{-1\leq \mathbf{l}\leq 1} \mathbf{l}^t\mathbf{y} \tag{6}$$

where vector $\mathbf{l}$ takes element-wise values between -1 to 1. Using eq. 6, we can express eq. 4 as

$$\begin{aligned} C(\mathbf{v_p}) &= \min_{\mathbf{v_p}} \{\|\mathbf{z_p}-\mathbf{v_p}\|_2^2+\lambda \max_{-1\leq\mathbf{l}\leq 1}\mathbf{l}^t\mathbf{M}\mathbf{v_p}\} \\ &= \min_{\mathbf{v_p}} \max_{-1\leq\mathbf{l}\leq 1} \{\|\mathbf{z_p}-\mathbf{v_p}\|_2^2+\lambda\,\mathbf{l}^t\mathbf{M}\mathbf{v_p}\} \end{aligned} \tag{7}$$

Using min-max theorem [29], we can represent eq. 7 as

$$C(\mathbf{v_p}) \;=\; \max_{-1\leq\mathbf{l}\leq 1} \min_{\mathbf{v_p}} f(\mathbf{v_p},\mathbf{l}) \tag{8}$$

where

$$f(\mathbf{v_p},\mathbf{l}) = \|\mathbf{z_p}-\mathbf{v_p}\|_2^2+\lambda\,\mathbf{l}^t\mathbf{M}\mathbf{v_p} \tag{9}$$

Solving the the minimization part of eq. 8, by performing $\frac{\partial}{\partial\mathbf{v_p}}f(\mathbf{v_p},\mathbf{l})=0$ and obtain

$$\mathbf{v_p} = \mathbf{z_p} - \frac{\lambda}{2}\mathbf{M}^t\mathbf{l} \tag{10}$$

Putting this value of $\mathbf{v_p}$ in eq. 8, we obtain,

$$C(\mathbf{v_p}) = \min_{-1\leq\mathbf{l}\leq 1} W(\mathbf{l}) \tag{11}$$

where $W(\mathbf{l}) = \mathbf{l}^t\mathbf{M}\mathbf{M}^t\mathbf{l} - \frac{4}{\lambda}\mathbf{l}^t\mathbf{M}\mathbf{z_p}$.
eq. 11 can be solved using generalized majorization-minimization (MM) algorithm [29] setting $\mathbf{l}^{(i)}$ as point of coincidence. If γ is a constant, then $\mathbf{G} = (\beta\mathbf{I}-\mathbf{M}\mathbf{M}^t)$ is a non-negative matrix that can construct the majorizer of $W(\mathbf{l})$, where $\mathbf{I}$ is an identity matrix of size $(N-1)\times(N-1)$.
Defining $i\geq 0$, $\mathbf{l}^{(0)}=0$ and $\gamma > maxeig(\mathbf{M}\mathbf{M}^t)$, the update term for $\mathbf{l}$ can be defined using MM algorithm as

$$\begin{aligned} \mathbf{l}^{(i+1)} &= \operatorname*{argmin}_{-1\leq\mathbf{l}\leq 1} \{W(\mathbf{l}) + (\mathbf{l}-\mathbf{l}^{(i)})^t(\gamma\mathbf{I}-\mathbf{M}\mathbf{M}^t)(\mathbf{l}-\mathbf{l}^{(i)})\} \\ &= \operatorname*{argmin}_{-1\leq\mathbf{l}\leq 1} \{\mathbf{l}^t\mathbf{r} - 2(\frac{1}{\gamma}\mathbf{M}(\frac{2}{\lambda}\mathbf{z_p}-\mathbf{M}^t\mathbf{l}^{(i)})+\mathbf{l}^{(i)})^t\mathbf{l}\} \\ &= \operatorname*{argmin}_{-1\leq\mathbf{l}\leq 1} \{\mathbf{l}^t\mathbf{l} - 2\mathbf{a}^t\mathbf{l}\} \end{aligned} \tag{12}$$

where

$$\mathbf{a} = \frac{1}{\gamma}\mathbf{M}(\frac{2}{\lambda}\mathbf{z_p}-\mathbf{M}^t\mathbf{l}^{(i)})+\mathbf{l}^{(i)}$$

We require to find $\mathbf{l}\in\mathbb{R}^N$, subject to $-1\leq l\leq 1$. By taking derivative of $\mathbf{l}^{(i+1)}$ with respect to $\mathbf{l}$ and equating to zero, it is can be shown that $l^{(i+1)}$ is minimized at $l_*^{(i)} = a$ if $|a|\leq 1$. In cases where $|a|>1$, $l^{(i+1)}$ is minimized at $l_*^{(i)} = sign(1)$ to satisfy the constraint $-1\leq r\leq 1$.

Finally we construct the two videos $\mathbf{L}$ and $\mathbf{S}$ where video $\mathbf{L}$ contains the background information of the input video and $\mathbf{S}$ contains the residual feature part of the input video $\mathbf{V}$ defined as $\mathbf{S}=\mathbf{V}-\mathbf{L}$, the intensity values at pixel location $\mathbf{p}$ in the i^{th} frame are v_p^i and s_p^i for videos $\mathbf{L}$ and $\mathbf{S}$ respectively.

B. Spline Interpolation of Features

Spline is a piecewise smooth function with minimum curvature property [24]. Conventionally, for a spline with degree n, each segment of the spline is represented using a nth degree polynomial, i.e., we need $(n+1)$ coefficients to represent each segment of a spline. In this paper we consider conventional B-spline model to interpolate the feature part. The core idea behind the model is that the background part of a video remains unchanged during an entire shot. Thus, temporally there is no change in data at the pixels that belongs to the background. The information in the feature video on the other hand will be sparse, and in most of the pixel locations do not require any interpolation. Thus, instead of interpolating the entire video directly, we interpolate at the pixel locations where the feature video has at least some non-zero components.

Suppose $\mathbf{V}$ is the input video consisting of frames $[f_1, f_2, ...f_N]$ with frame dimension $M\times N$.
We have assumed each pixel as knot to fit the Cubic Spline, as then we have selected one particular pixel position in all frames and assumed all those pixels as numbers in an array with the frame number as abscissa and pixels value as ordinate or value of the function at that point and then fitted Cubic Spline between each knot (i.e., between two abscissas), as an example suppose pixel $\mathbf{p}$ of all the frames is selected, i.e, $f_1(\mathbf{p}), f_2(\mathbf{p}), \ldots, f_n(\mathbf{p})$ and assumed each pixel position $(f_1(\mathbf{p}),\ldots,f_n(\mathbf{p}))$ as the abscissas and the intensity values of the pixels as the function's value, and then we fit a Cubic spline between each knot.
To reduce the time complexity of Spline fitting, we have implemented the Cubic Spline on the sparse part of video. Input video V decomposed into sparse video $\mathbf{S}$ and background video $\mathbf{L}$ such that the rank of $\mathbf{L}$ is 1.
First, for a particular pixel $\mathbf{p}$, we construct a vector $\mathbf{s_p}$ of length $N+1$ from video $\mathbf{S}$ such that

$$s_p^i = \begin{cases} S_i(\mathbf{p}); & i = 1,2,\ldots N \\ S_N(\mathbf{p}); & i = N+1 \end{cases} \tag{13}$$

where s_p^i is the i^{th} element of vector $\mathbf{s_p}$ and $S_i(\mathbf{p})$ is the intensity value at pixel location $\mathbf{p}$ in the i^{th} frame of video $\mathbf{S}$.

We perform the spline interpolation at pixel $\mathbf{p}$ only if $\|\mathbf{s_p}\|_1 > \tau$. In ideal condition, i.e., in absence of noise, τ should be equal to zero. However, in practical cases, an input video has different types of noise and we consider $\tau = \mu + \sigma$ where μ and σ are the mean and standard deviation of the absolute value of video $\mathbf{S}$ when represented as a column vector. To up-sample a video by a factor l, first we generate $(l-1)$ number of points between s_p^i and s_p^{i+1}. Thus, after upsampling, we have lN total number of data points- $(l-1)N$ number of interpolated points and N number of original data point, neglecting the point s_p^N. If this upsampled vector is $\mathbf{s_{p_u}}$ Then we define a video $\mathbf{S^u}$ such that

$$S_i^u(\mathbf{p}) = \begin{cases} s_{p_u}^i & \text{if } \|\mathbf{s_p}\|_1 > \tau \\ 0 & \text{if } \|\mathbf{s_p}\|_1 \leq \tau \end{cases} \quad (14)$$

where $S_i^u(\mathbf{p})$ is the intensity value at pixel location $\mathbf{p}$ in the i^{th} frame of video $\mathbf{S^u}$ and $s_{p_u}^i$ is the i^{th} element of vector $\mathbf{s_{p_u}}$.

We also defined the upsampled background video $\mathbf{L^u}$ as

$$L_i^u = L_1; \quad i = 1, 2, \ldots lN \quad (15)$$

where L_i^u is the i^{th} frame of video $\mathbf{L^u}$ and L_1 is the first frame of video $\mathbf{L}$. It is important to note that as the rank of video $\mathbf{L}$ is 1, all the frames are visually similar in $\mathbf{L}$. Thus, instead of selecting the first frame of video $\mathbf{L}$ in eq. 15, we can select any frame of video $\mathbf{L}$.

Finally, we construct the upsampled video $\mathbf{V^u}$ as

$$\mathbf{V^u} = \mathbf{L^u} + \mathbf{S^u} \quad (16)$$

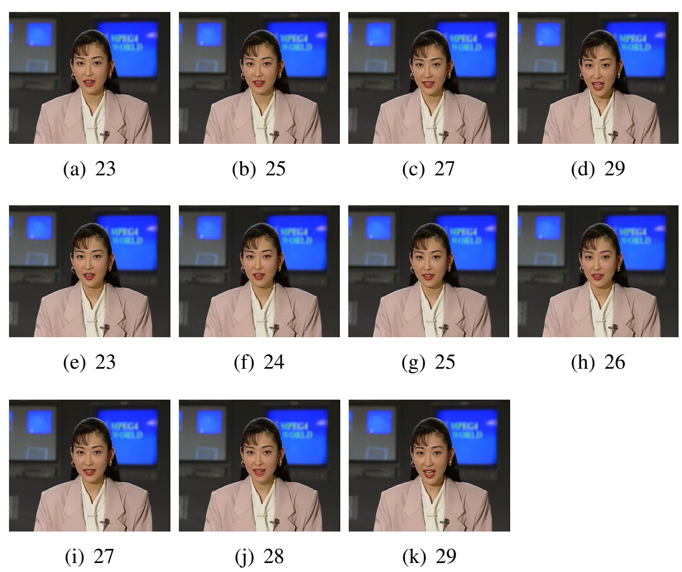
(a) 20 (b) 22 (c) 24 (d) 26
(e) 20 (f) 21 (g) 22 (h) 23
(i) 24 (j) 25 (k) 26

Fig. 1: Output of the proposed algorithm on Mother dataset: (a) input frames (from a20 to d26, where a20, b22, c24 and d26 represents frame no. 20, 22, 24, 26); (b) output frames (e20 to k26, where e20, f22, g22, h23, i24, j25 and k26 represents frame no. 20, 21, 22, 23, 24, 25, 26)of the interpolated video

III. Experimental Result

The proposed method was compared with the other three existing interpolation methods. These methods are Multi-channel Mixed Pattern Based Frame Rate up conversion (MCMP-FRUC) [21], direction-select ME(DS-ME) [20] and extended bilateral motion estimation (EBME) [11]. We test the algorithms on several standard video datasets like *Akiyo, Mother, Carphone, Foreman, Bus, Flower* etc., and use peak signal to noise ratio (PSNR) and structural similarity index (SSIM)as statistical measures to evaluate the quality of interpolated videos. In Fig. 1(a), we show the frames of Carphone dataset used for upconversion, and in Fig. 1(b), we show the frames after the upsampling of the video using the proposed method. In Fig. 2, we show the change of the metrics at each frame for the existing methods and the proposed method. It can be seen that the proposed method outperforms other three existing methods. We tabulate the average peak signal to noise ratio (APSNR) and average SSIM (ASSIM) for all the dataset in Table 1.

For a visual comparison, we show the interpolated videos to 25 different viewers and ask for their opinions about the perceptual quality of the upsampled videos. The mean opinion scores (MOS) of the viewers are shown in FIg.3, where score 10 and score 0 indicate highest quality and lowest quality of



(a) 23 (b) 25 (c) 27 (d) 29
(e) 23 (f) 24 (g) 25 (h) 26
(i) 27 (j) 28 (k) 29

Fig. 2: Output of the proposed algorithm on Akiyo Dataset: (a) input frames (from a23 to d29, where a23, b25, c27 and d29 represents frame no. 23, 25, 27, 29); (b) output frames (e23 to k29, where e23, f24, g25, h26, i27, j28 and k29 represents frame no. 23, 24, 25, 26, 27, 28, 29)of the interpolated video

TABLE I: Quantitative analysis of different upsampling algorithms

Dataset	Metrics	[21]	[20]	[11]	Proposed
Akiyo	APSNR(dB)	46.74	44.83	46.42	48.16
	ASSIM	0.971	0.992	0.991	0.998
Mother	APSNR(dB)	43.74	43.14	42.73	50.83
	ASSIM	0.851	0.991	0.986	0.997
Carphone	APSNR(dB)	34.70	33.66	34.18	45.29
	ASSIM	0.847	0.961	0.964	0.993
Foreman	APSNR(dB)	40.25	40.08	39.17	47.29
	ASSIM	0.851	0.947	0.939	0.995
Bus	APSNR(dB)	44.61	43.93	43.18	59.83
	ASSIM	0.834	0.926	0.911	0.991
Flower	APSNR(dB)	42.71	41.88	41.62	49.13
	ASSIM	0.859	0.943	0.927	0.991

(a)

(b)

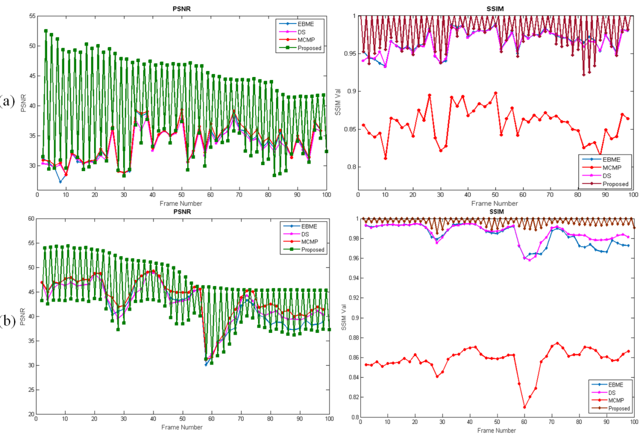


Fig. 3: PSNR and SSIM of each frame for (a) Carphone Dataset; (b) Mother Dataset using MCMP [21], DS [20], EBME [11] and proposed algorithm.

a video according to a viewer. It can be observed that MCMP performs well in almost all the dataset, except *Bus* dataset where large motion is present. EBME performs poorly if the scene complexity is high. The proposed algorithm, however performs well in all the test conditions.

In Table 1 we used quantifying measures like PSNR and SSIM which requires reference image to compare the quality of the interpolated image. But its not always possible to have reference image for quality comparison. Hence, in Table 2 we used non reference based image quality measurement techniques like BRISQUE [30], ILNIQE [31], etc. along with some correlation and structural based quantifying measures like structural content, visual signal to noise ratio , etc. that requires reference image.

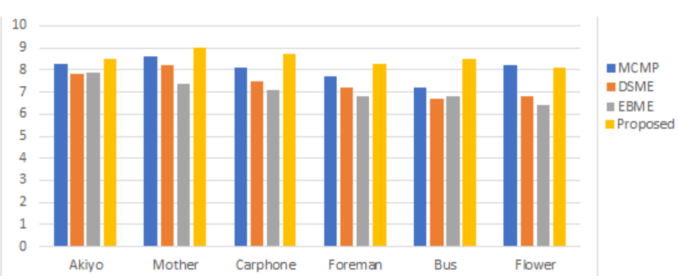


Fig. 4: Mean opinion scores of existing algorithms and the proposed method for different input videos.

TABLE II: Quantitative analysis of different upsampling algorithms

Metrics	Datasets	[21]	[20]	[11]	Proposed
Absolute Average Difference	Akiyo	3.041	3.0476	3.011	0.0693
	Carphone	2.672	2.6747	2.661	0.2463
	Mother	2.229	2.2190	2.216	0.0984
BRISQUE [30]	Akiyo	19.281	19.401	19.244	17.342
	Carphone	36.022	37.648	35.830	33.209
	Mother	18.281	18.709	17.966	15.122
ILNIQE [31]	Akiyo	30.951	31.192	30.921	27.569
	Carphone	57.251	58.169	57.278	54.252
	Mother	72.692	73.454	72.547	69.661
JPEG [32]	Akiyo	31.783	31.760	31.762	28.877
	Carphone	38.745	39.278	38.613	35.192
	Mother	27.321	27.394	27.218	24.392
MSE	Akiyo	21.427	21.739	21.611	10.482
	Carphone	50.372	53.212	52.723	47.121
	Mother	20.071	21.962	21.726	17.368
Normalized Absolute Error	Akiyo	0.041	0.0402	0.0402	0.0198
	Carphone	0.044	0.0455	0.0451	0.0329
	Mother	0.027	0.0293	0.0294	0.0201
Structural Content [35]	Akiyo	0.9331	0.9393	0.9392	0.9995
	Carphone	0.9521	0.9582	0.9573	0.9992
	Mother	0.9661	0.9633	0.9631	0.9996
Universal Image Quality Index [33]	Akiyo	0.9751	0.9858	0.9771	0.9994
	Carphone	0.9822	0.9895	0.9885	0.9993
	Mother	0.9881	0.9891	0.9886	0.9999
Visual Signal to Noise Ratio [34]	Akiyo	36.78	36.61	36.64	39.95
	Carphone	18.57	18.53	18.56	20.84
	Mother	9.56	9.81	9.77	12.64

To measure Absolute Average Difference(AAD), we calculated the absolute difference between the pixels value of original image and interpolated image and then normalized the result by the size of image. Structural Content (SC) [35] is correlation based measure and it measures the similarity between images. To calculate SC, we divided the sum of the squares of interpolated frame pixels to sum of the squares of original frame pixels. To calculate Normalized Absolute Error (NAE), we divided the absolute difference between the pixels value of original image and interpolated image and normalized by the sum of the pixel value of original Image. For AAD and NAE, lower value implies better interpolated frames and for SC, higher value indicates better interpolated frames. To measure Universal Image quality index (UIQI), we defined a term quality index Q as mentioned in [33], and measured its value. The value of Q can vary dynamically in range of [-1, 1], 1 is the best. In Visual Signal to Noise ratio(VSNR) method [34], we operated in two-steps. Firstly, we computed the threshold value for detecting distortion in interpolated images by using natural images in wavelet-based model. If the distortions are below the threshold of detection, then value of VSNR will be high and no further analysis is required. If the distortions are above threshold, then VSNR is computed using linear sum of Euclidean distances between two HVS property modeled as, low-level visual property of perceived contrast, and the mid-level visual property of global precedence [34]. In VSNR values are measured in db, high value of VSNR indicates better interpolated frames. Above, discussed techniques require reference image to get results. We have also used non-reference based image quality techniques. In BRISQE [30], point wise statistics of local normalized luminance was extracted to evaluate the interpolated frames.

Since in BRISQE, we calculated the deviation of interpolated image from natural image model so lower value of quality score indicates better interpolated frames.

In ILNIQE [31], we learned a generative model called multivariate Gaussian model using image patches of naural images used as training data. To calculate the quality score of each image, we first measured the Bhattacharyya distance for each image patches, then an overall quality score is obtained by taking the mean of each patch value as stated in [31]. In ILNIQE also, we calculated the quality score in terms of deviation so lower value of quality score implies better interpolated images.

In JPEG [32], we have used regression model to define quality score S where,

$$\mathbf{S} = \alpha + \beta B^{\gamma_1} A^{\gamma_2} Z^{\gamma_3}$$

where, A represents average absolute difference between in block-samples, difference across block boundaries is represented by B, zero-crossing rate is represented as Z and α, β, γ_1, γ_3 and γ_3 are the model parameters that we calculated using test data and non-linear regression method. Higher value of S indicates better quality of interpolated frame.

IV. Conclusion

In this paper, we proposed a novel up-conversion algorithm for video data. Instead of global upsampling, we perform the interpolation locally to save processing time and it also produces less distortion. Another advantage of the proposed algorithm is that the entire operation is pixel-based. Thus, both the processing steps- decomposition and interpolation, both are parallelizable, and processing time can be further reduced in multi-core processing environment. The proposed algorithm better preserves the color and structural information than the existing up-conversion algorithms as shown in the experimental results. It is evident that the algorithm works well when the upsampling factor $l \in \mathbb{Z}_+$, where $\mathbb{Z}_+$ is the positive integer space. However, in many cases, the sample rate conversion requires non-integer upsampling factor. In future, we will work to develop interpolation method for non-integer upsampling factor.

References

[1] J. Someya, N. Okuda and H. Sugiura, The Suppression of Noise on a Dithering Image in LCD Overdrive, IEEE Trans. Consumer Electronics, vol. 52, no. 4, pp. 1325-1332, Nov. 2006.

[2] Y.-H. Ko, J.-H. Kim, S.-W. Lee, and H.-S. Kang, Dual Block Truncation Coding for Overdriving of Full HD LCD Driver, IEEE Trans. Consumer Electronics, vol. 58, no. 1, pp.1-7, Feb. 2012.

[3] S. H. Chan, T. X. Wu, and T. Q. Nguyen, Comparison of Two Frame Rate Conversion Schemes for Reducing LCD Motion Blurs, IEEE Signal Processing Letters, vol. 17, no. 9, Sept. 2010.

[4] S. J. Kang and Y. H. Kim, Performance comparison of motion estimation methods for frame rate up-conversion, Proc. International Display Workshops (IDW), Otsu, Japan, pp. 1633-1636, Dec. 2006

[5] T. Y. Kuo and C.-C. J. Kuo, Motion-compensated interpolation for low-bit-rate video quality enhancement, Proc. SPIE Applications of Digital Image Processing, San Diego, USA, vol. 3460, pp. 277288, Jul.1998.

[6] A. Kaup and T. Aach, Efficient prediction of uncovered background in interframe coding spatial extraprediction, Proc. International Conference on Acoustics, Speech and Signal Processing, Adelaide, Australia, vol. 5, pp. 501-504, Apr. 1994.

[7] P. Blanchfield, D. Wang, A. Vincent, F. Speranza, and R. Renaud, Advanced frame rate conversion and performance evaluation, SMPTE Motion Imag. J., pp. 153159, Apr. 2006.

[8] B.-W. Jeon, G.-I. Lee, S.-H. Lee, and R.-H. Park, Coarse-to-fine frame interpolation for frame rate up-conversion using pyramid structure, IEEE Trans. Consumer Electronics, vol. 49, no. 3, pp. 499508, Aug. 2003.

[9] B. T. Choi, S. H. Lee, and S. J. Ko, New frame rate up-conversion using bidirectional motion estimation, IEEE Trans. Consumer Electronics, vol. 46, no. 3, pp. 603-609, Aug. 2000.

[10] B.-D. Choi, J.-W. Han, C.-S Kim, and S.-J. Ko, Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation, IEEE Tran. Circuits and Systems for Video Technology, vol. 17, pp. 407-416, Apr. 2007.

[11] S.-J. Kang, K.-R. Cho, and Y. H. Kim, Motion Compensated Frame Rate Up-Conversion Using Extended Bilateral Motion Estimation, IEEE Tran. Consumer Electronics, vol. 53, no. 4, pp. 1759-1767, Nov. 2007.

[12] S.-J. Kang, S.-K. Lee, and Y. H. Kim Multiframe-based Bilateral Motion Estimation with Emphasis on Stationary Caption Processing for Frame Rate Up-Conversion, IEEE Trans. Consumer Electronics, vol. 54, no. 4, pp. 18301838, Nov. 2008.

[13] J.-H. Kim, Y.-H. Ko, H.-S. Kang, S.-W. Lee, and J. W. Kwon, Frame Rate Up-Conversion Method Based on Texture Adaptive Bilateral Motion Estimation, IEEE International Conference on Consumer Electronics, Las Vegas, USA, pp. 107-108, Jan. 2014.

[14] T. Liu, K.-T. Lo, J. Feng, and X. Zhang, Frame interpolation scheme using inertia motion prediction, Signal Process.: Image Commun., vol. 18, pp. 221229, 2003.

[15] S. Liu, C.-C. J. Kuo, and J. W. Kim, Hybrid global-local motion compensated frame interpolation for low bit rate video coding, J. Visual Commun. Image Represent., vol. 14, pp. 6179, 2003.

[16] D. Wang, A. Vincent, P. Blanchfield, and R. Klepko, Motion Compensated Frame Rate Up-Conversion-Part II: New Algorithms for Frame Interpolation, IEEE Transactions on Broadcasting, vol.56, no.02, pp.142-149, 2010.

[17] D. Wang, L. Zhang, and A. Vincent, Motion Compensated Frame Rate Up-Conversion-Part I: Fast Multi-Frame Motion Estimation, IEEE Transactions on Broadcasting, vol.56, no.2, pp.133-141, 2010.

[18] S. Kang, S. Yoo and Y. Kim, Dual Motion Estimation for Frame Rate Up Conversion, IEEE Transactions on Circuits System Video Technology, vol.20, no.12, pp. 1909-1914, 2010.

[19] Yehuda Dar and Alfred M. Bruckstein, Motion-Compensated Coding and Frame Rate Up-Conversion: Models and Analysis, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 24, NO. 7, JULY 2015.

[20] Dong-Gon Yoo, Suk-Ju Kang, and Young Hwan Kim, Direction-Select Motion Estimation for Motion-Compensated Frame Rate Up-Conversion, JOURNAL OF DISPLAY TECHNOLOGY, VOL. 9, NO. 10, OCTOBER 2013.

[21] Ran Li, Zongliang Gan, Ziguan Cui, Guijin Tang, and Xiuchang Zhu, Multi-Channel Mixed-Pattern Based Frame Rate Up-Conversion Using Spatio-Temporal Motion Vector Refinement and Dual-Weighted Overlapped Block Motion Compensation, JOURNAL OF DISPLAY TECHNOLOGY, VOL. 10, NO. 12, DECEMBER 2014.

[22] I.J. Schoenbcrg, Contribution to the problem of approximation of equidistant data by analytic functions, Quart.Appl.Math., vol. 4, pp, 45-99, 112-141, 1946

[23] Hsieh S. Hou, Harry C. Andrews, Cubic Splines for image Interpolation and Digital Filtering, IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, VOL. ASSP-26, N0.6, DECEMBER 1978.

[24] T. K. Truong, L. J. Wang, I. S. Reed, and W. S. Hsieh, Image Data Compression Using Cubic Convolution Spline Interpolation, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 9, NO. 11, NOVEMBER 2000.

[25] Lung-Jen Wang, Wen-Shyong Hsieh, Trieu-Kien Truong, Irving S. Reed, and T. C. Cheng,A Fast Efficient Computation of Cubic-Spline Interpolation in Image Codec, IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 49, NO. 6, JUNE 2001.

[26] Robert G. Keys, Cubic Convolution Interpolation for Digital Image Processing, IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, VOL. ASSP-29, NO. 6, DECEMBER 1981.

[27] Tom Lyche, Discrete cubic spline interpolation, BIT Numerical Mathematics Volume 16, Issue 3, pp 281-290, September 1976.

[28] Donoho, David L. Compressed sensing. IEEE Transactions on information theory 52.4 (2006): 1289-1306.

[29] Selesnick, Ivan W., and Ilker Bayram. Total variation filtering. White paper (2010).
[30] A. Mittal, A. K. Moorthy and A. C. Bovik, No Reference Image Quality Assessment in the Spatial Domain, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 21, NO. 12, DECEMBER 2012.
[31] Lin Zhang, Lei Zhang, and A.C. Bovik, A feature-enriched completely blind image quality evaluator, IEEE TRANSACTIONS ON IMAGE PROCESSING, 2015.
[32] Zhou Wang, Hamid R. Sheikh and Alan C. Bovik, "No-Reference Perceptual Quality Assessment of JPEG Compressed Images," submitted to IEEE International Conference on Image Processing, Sept. 2002.
[33] A Universal Image Quality Index by Zhou Wang and Alan C. Bovik, published in IEEE Signal Processing Letters, 2002.
[34] Chandler, D. M., Hemami, S. S. "VSNR: A Wavelet-Based Visual Signal-to-Noise Ratio for Natural Images", IEEE Trans Image Processing, 16(9):2284 - 2298, 2007.
[35] C.Sasi varnan, A.Jagan, Jaspreet Kaur, Divya Jyoti, Dr.D.S.Rao, Image Quality Assessment Techniques pn Spatial Domain, ISSN : 2229-4333(Print) — ISSN : 0976-8491(Online) www.ijcst.com, IJCST Vol. 2, Issue 3, September 2011
[36] C.C. Chang and C.J. Lin. LIBSVM: A library for support vector machines. 2001.
[37] Fredrik Vestermark, Implementation of a Frame Rate Up-Conversion Filter, Mater's Thesis in computer science, 2013.