Compression of Temporal Video Data by Catmull-Rom Spline and Quadratic Bézier Curve Fitting

Murtaza Khan and Yoshio Ohno

WSCG 2008

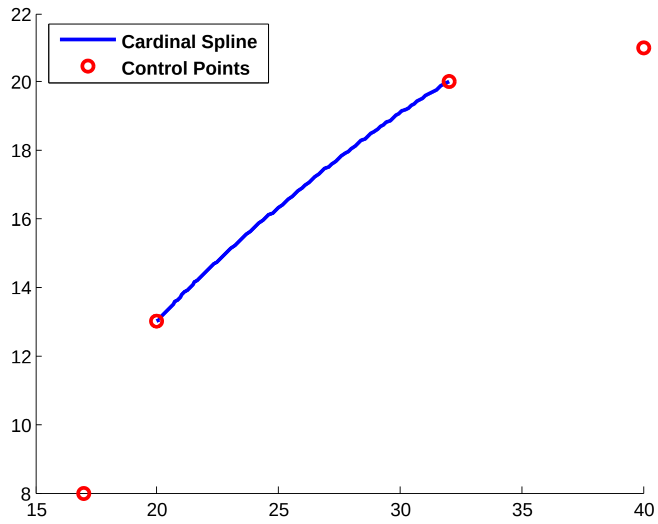
January 19, 2008

Ohno Lab
Graduate School of Science & Technology
Keio University

Contents

1. Catmull-Rom Spline (CRS)

- Catmull-Rom Spline is a C^1 continuous curve. Cardinal spline interpolates piecewise cubics for each segment.
- A CRS segment is defined by four control points, i.e., P_{j-1}, P_j, P_{j+1} and P_{j+2}.
- The j^{th} segment of Cardinal spline interpolates between two *middle control points*, i.e., P_j and P_{j+1}. The *end control points*, i.e., P_{j-1} and P_{j+2} are used to calculate the tangent of P_j and P_{j+1}.

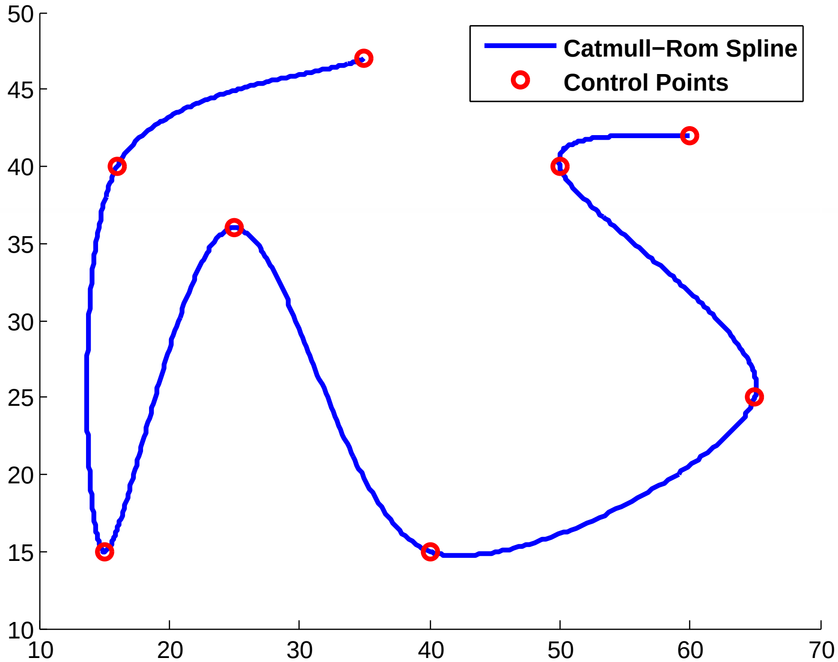


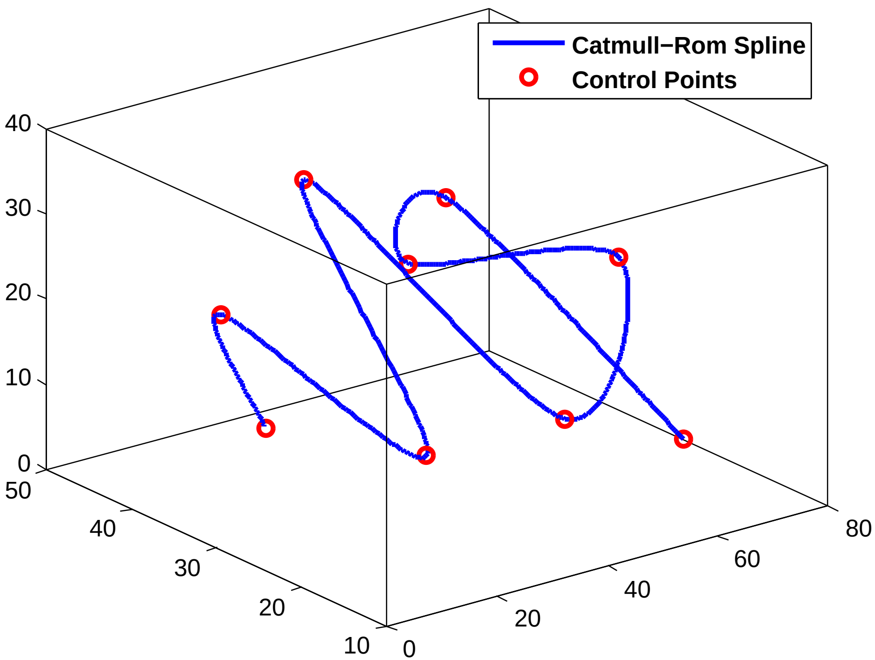
Catmull-Rom Spline

$$\begin{aligned} Q_j(t_i) =& \frac{1}{2}[(-t_i^3 + 2t_i^2 - t_i)P_{j-1} \\ & + [3t_i^3 - 5t_i^2 + 2]P_j \\ & + [-3t_i^3 + 4t_i^2 + t_i]P_{j+1} \\ & + (-t_i^3 - t_i^2)P_{j+2}], \end{aligned} \tag{1}$$

where t_i is parameter of interpolation , $0 \leq t_i \leq 1$. In order to generate n points between P_j and P_{j+1} inclusive, the parameter t_i is divided into $(n-1)$ intervals between 0 and 1 inclusive, and $Q_j(t_i)$ is evaluated at n values of t_i.

KEIO UNIVERSITY 1858



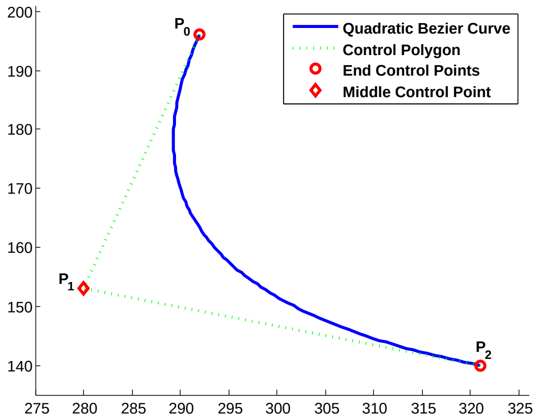


2. Quadratic Bézier Curve (QBC)

- Quadratic Bézier curve (QBC) is a C^0 continuous curve.
- A QBC segment, is defined by three control points, i.e., P_0, P_1, and P_2. P_0 and P_2 are called *end control points* (*ECP*), while P_1 called a *middle control point* (*MCP*).
- To generate continuous QBC that interpolate $k+1$ points k curve segments are used. Equation of a QBC segment can be written as follows:

$$Q(t_i) = (1-t_i)^2 P_0 + 2t_i(1-t_i)P_1 + t_i^2 P_2, \tag{2}$$

where t_i is a parameter of interpolation , $0 \leq t_i \leq 1$. In order to generate n points between P_0 and P_2 inclusive, the parameter t_i is divided into $(n-1)$ intervals between 0 and 1 inclusive, and $Q(t_i)$ is evaluated at n values of t_i.



3. QBC Least Square Fitting

Middle control point (*MCP*) i.e., P_1 of QBC is obtained by least square method. Least square method gives the *best* value of *MCP* that minimizes the squared distance between the original and the fitted data. If there are m data points in a segment, and O_i and $Q(t_i)$ are values of original and approximated points respectively then we can write the least square equation as follows:

$$U = \sum_{i=1}^{m} [O_i - Q(t_i)]^2. \tag{3}$$

Substituting the value of $Q(t_i)$ from Eq. (2) in Eq. (3) yields:

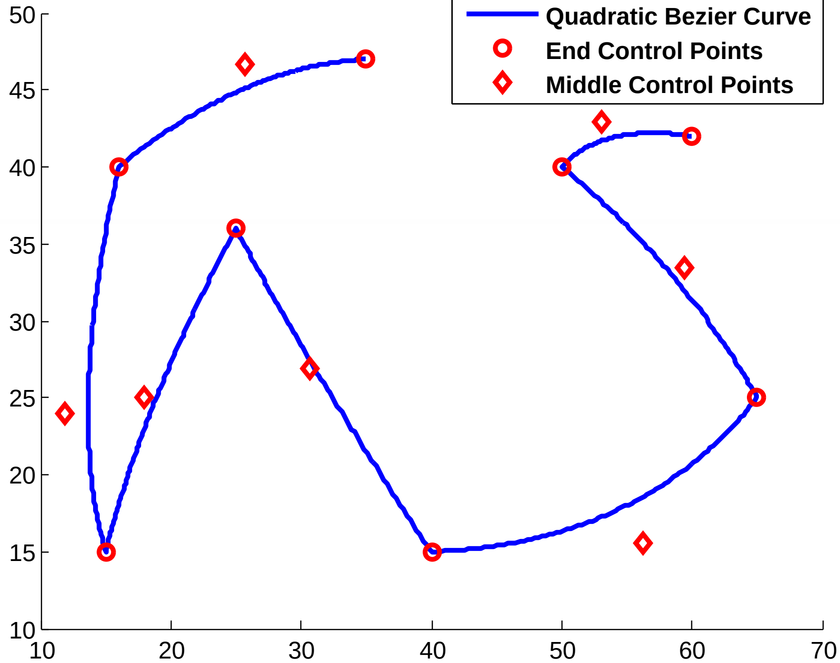
$$U = \sum_{i=1}^{m} [p_i - (1-t_i)^2 P_0 + 2t_i(1-t_i)P_1 + t_i^2 P_2]^2. \tag{4}$$

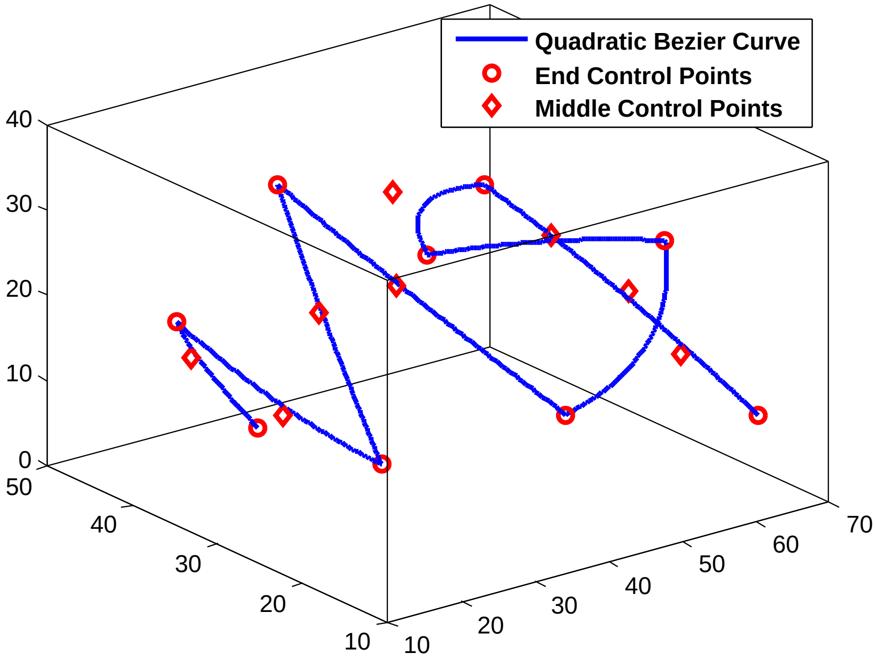
Differentiating Eq. (4) partially with respect to P_1 yields:

$$\frac{\partial U}{\partial P_1} = 0. \tag{5}$$

Solving Eq. (5) for P_1 gives:

$$P_1 = \frac{\sum_{i=1}^{m} \left[p_i - (1-t_i)^2 P_0 - t_i^2 P_2\right]}{\sum_{i=1}^{n} 2t_i(1-t_i)}. \tag{6}$$


Quadratic Bezier Curve
End Control Points
Middle Control Points
50
45
40
35
30
25
20
15
10
10
20
30
40
50
60
70

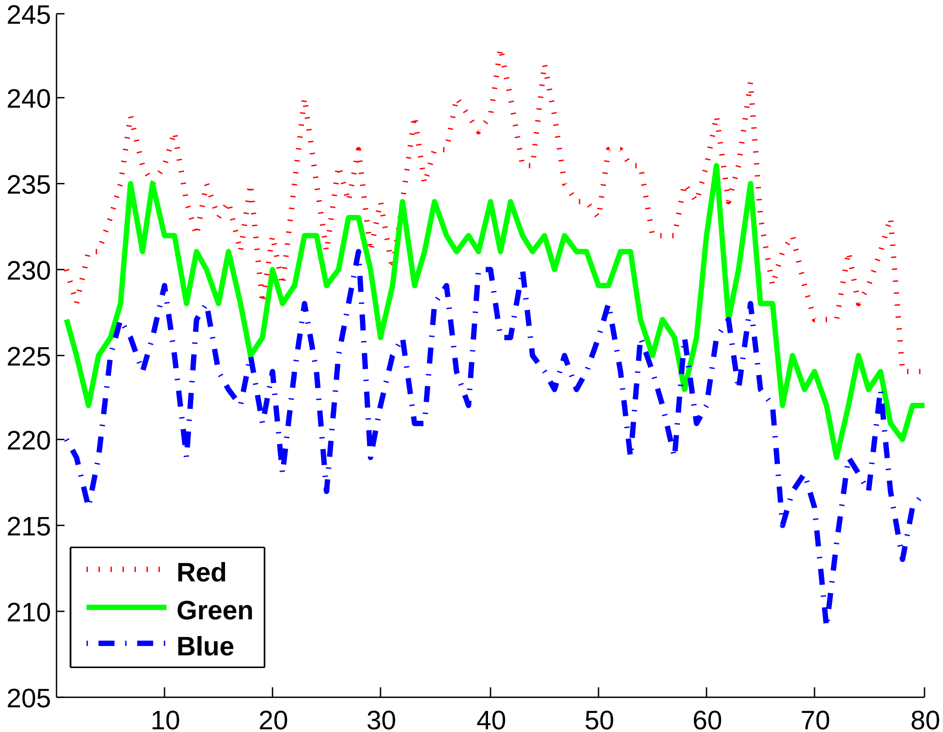

Quadratic Bezier Curve
End Control Points
Middle Control Points
40
30
20
10
0
50
40
30
20
10
10
20
30
40
50
60
70

4. Video Data Compression

- Prevalent temporal video data compression methods are are called motion estimation (ME) or motion compensation (MC) methods.
- ME algorithms are based on temporal changes in intensities of sequence of frames.
- It is quite possible that there is change in intensities without actual motion e.g., camera movements or illumination conditions changes.
- We developed a method of lossy temporal video data compression using spline fitting.
- Spline based intensity approximation methods are more robust because they work in both situations i.e., changes in intensities with or without actual motion. Whereas conventional motion compensation methods based on block matching are dependent on actual motion of object (block) to find the matching block.

Video Data Compression

- Digital video data consists of sequence of frames (images) in temporal dimension. Each frame consists of rectangle 2D array of pixels.
- Intensity or color values are associated with each pixel.
- Value of a pixel in a frame can be considered as a point in Euclidean space R^1 or R^3 for intensity and color respectively.
- If a video consists of a sequence of M frames then for each pixel we have a set of values $\{p_1, p_2, \ldots, p_M\}$, i.e., $p_j = I_j$ or $p_j = (X_j, Y_j, Z_j)$, where $1 \leq j \leq M$. I is intensity and XYZ can be pixel values in RGB, YC_bC_r or HSV color space.



- *RGB* temporal variation of a pixel in 80 frames of a video.

5. Fitting Strategy

- Fitting process is applied to temporal data of each pixel individually. We have to approximate the M values of each pixel $O = \{p_1, p_2, \ldots, p_M\}$ (original data) by spline fitting.
- Input: (1) *upper limit of error* ξ^{lmt}, i.e., maximum allowed square distance between original and fitted data, e.g., $\xi^{lmt} = 100$ (2) *initial breakpoint interval* Δ, i.e., pixel after every Δ^{th} frames is taken as an *end control point*, e.g., $\Delta = 12$ then set of initial keypixels K is $K = \{p_1, p_{13}, p_{25}, p_{37}, \ldots, p_n\}$.
- The fitting process divides the data into segments based on keypixels. A segment is set of all points (pixels) between two consecutive keypixels, e.g., $S_1 = \{p_1, p_2, \ldots, p_{13}\}$, $S_2 = \{p_{13}, p_{14}, \ldots, p_{25}\}$.

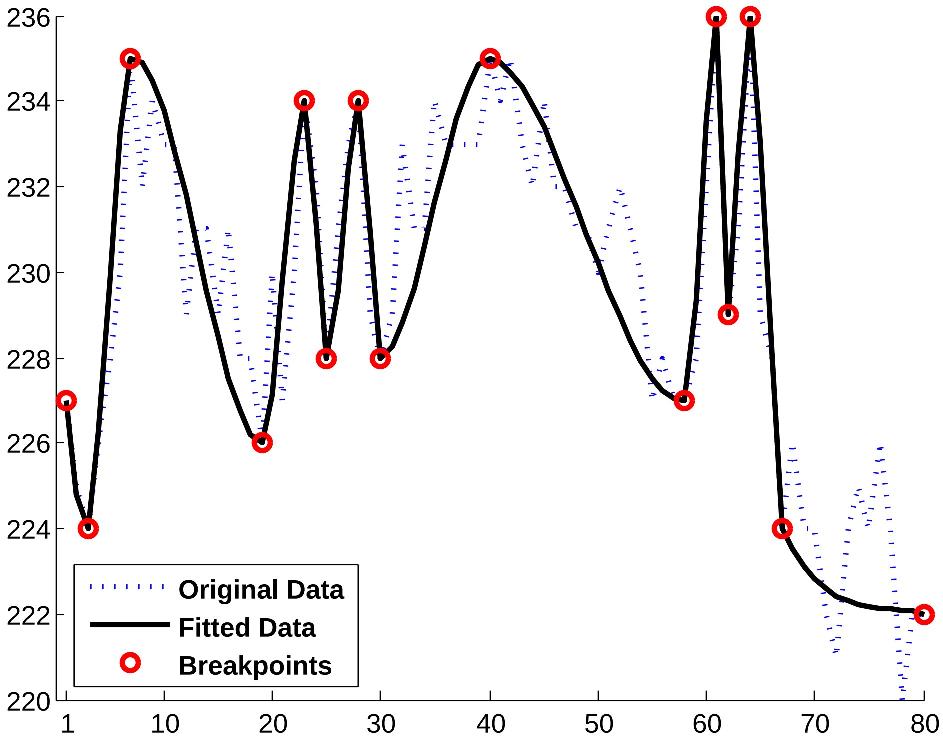
6. Fitting Strategy

- Fitting process is applied to temporal data of each pixel individually. Color or luminance value of a pixel at frame i is p_i, where $0 \leq p_i \leq 255$ and $1 \leq i \leq n$. We have to approximate the n values of each pixel $O = \{p_1, p_2, \ldots, p_n\}$ (original data) by spline fitting.
- Input: (1) *upper limit of error* ξ^{lmt}, i.e., maximum allowed square distance between original and fitted data, e.g., $\xi^{lmt} = 100$ (2) *initial keyframe interval* Δ, i.e., pixel after every Δ^{th} frames is taken as an *end control point* of CRS or QBC, e.g., $\Delta = 12$ then set of initial keypixels K is $K = \{p_1, p_{13}, p_{25}, p_{37}, \ldots, p_n\}$.
- The fitting process divides the data into segments based on keypixels. A segment is set of all points (pixels) between two consecutive keypixels, e.g., $S_1 = \{p_1, p_2, \ldots, p_{13}\}$, $S_2 = \{p_{13}, p_{14}, \ldots, p_{25}\}$.

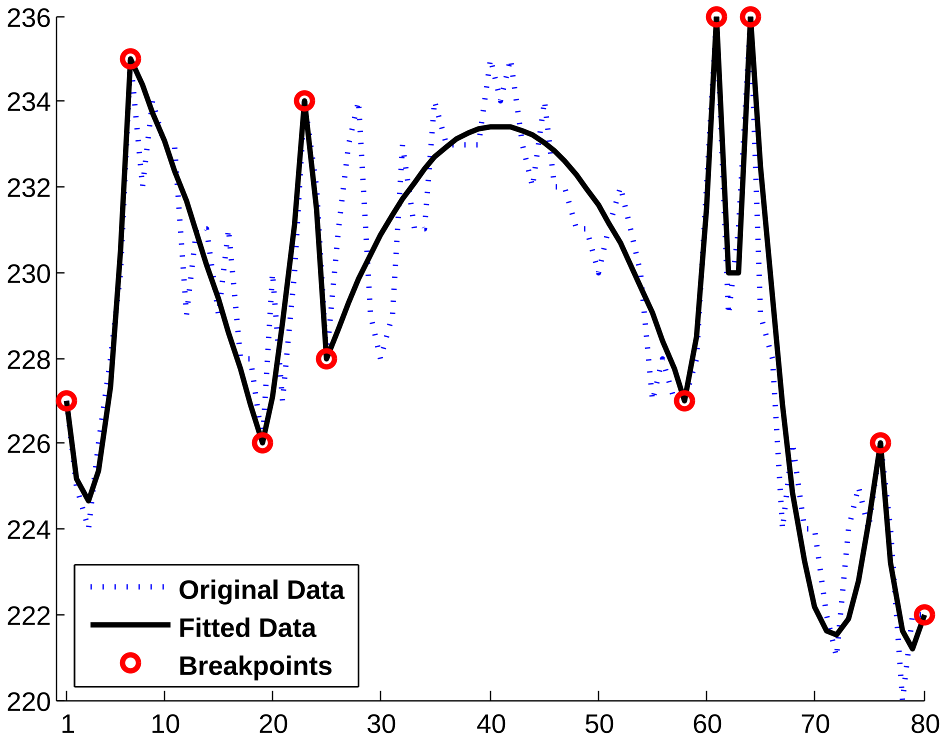
Fitting Strategy

- Spline interpolation is performed for each segment and n interpolated values (approximated data) $Q = \{q_1, q_2, \ldots, q_n\}$ are obtained.
- Then we compute the error, i.e., squared distance between each point of original data and its corresponding point on approximated data $d_i^2 = \|p_i - q_i\|^2$, $1 \le i \le n$. Among all error values we compute the maximum error $\xi^{max} = Max\left(d_1^2, d_2^2, \ldots, d_n^2\right)$. If ξ^{max} is greater than ξ^{lmt} then we add a point (new keyframe) from original data in the set of keyframes where the error is maximum between original and approximated data.
- Due to addition of a new keyframe a segment is split and replaced by two new segments. For example if $\xi^{max} = d_6^2$ then segment S_1 is split and a new keyframe 6 is inserted between keyframes 1 and 13 ($K = \{1, 6, 13, 25, 37, \ldots, n\}$) and two new segments $\{p_1, p_2, \ldots, p_6\}$ and $\{p_6, p_7, \ldots, p_{13}\}$ replace S_1. The fitting process is repeated with new set of keyframes until ξ^{max} is less than or equal to ξ^{lmt} for each segment.

KEIO UNIVERSITY



Catmull-Rom spline fitting to luminance values in 80 frames of a video, $\xi^{lmt} = 21$, $\delta = 79$, PSNR=43.845-dB.



Quadratic Bézier curve fitting to luminance values in 80 frames of a video, $\xi^{lmt} = 21$, $\delta = 79$, PSNR=45.115-dB.

KEIO UNIVERSITY

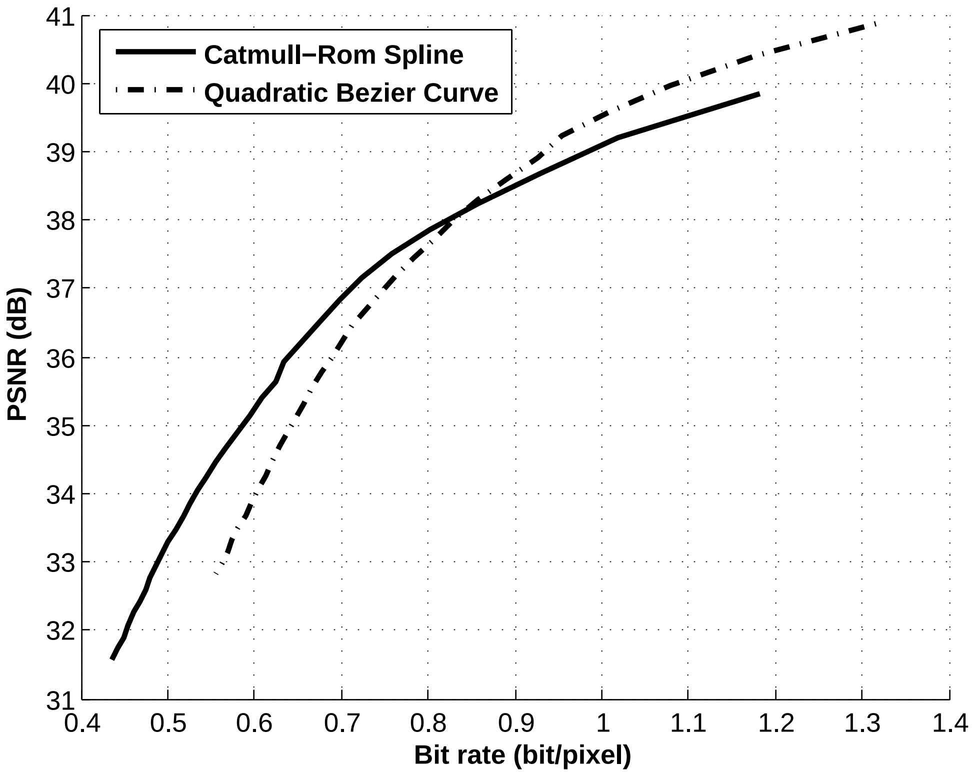
7. Experiments and Results

Details of input video sequences.

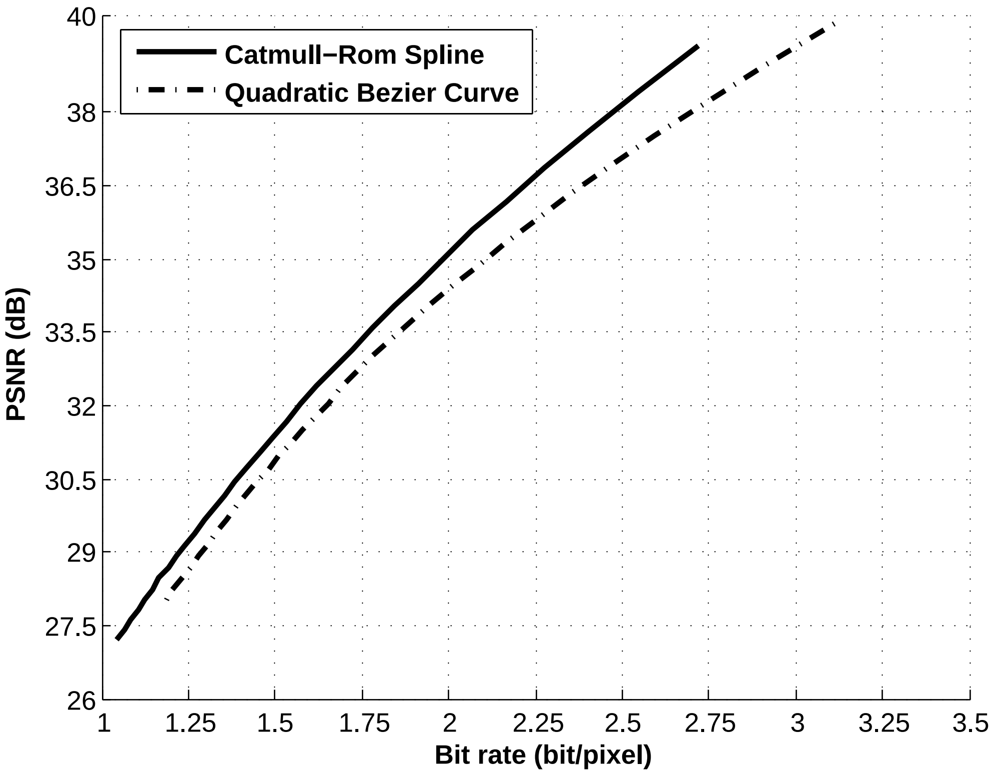
Video Name	Format	Number of Frames	Bit-rate
Salesman (luminance)	CIF 352×288	45	8-bpp
Foreman (luminance)	SIF 352×288	44	8-bpp
Darius (RGB)	SIF 352×288	44	24-bpp

Performance comparison of TSS, CRS and QBC.

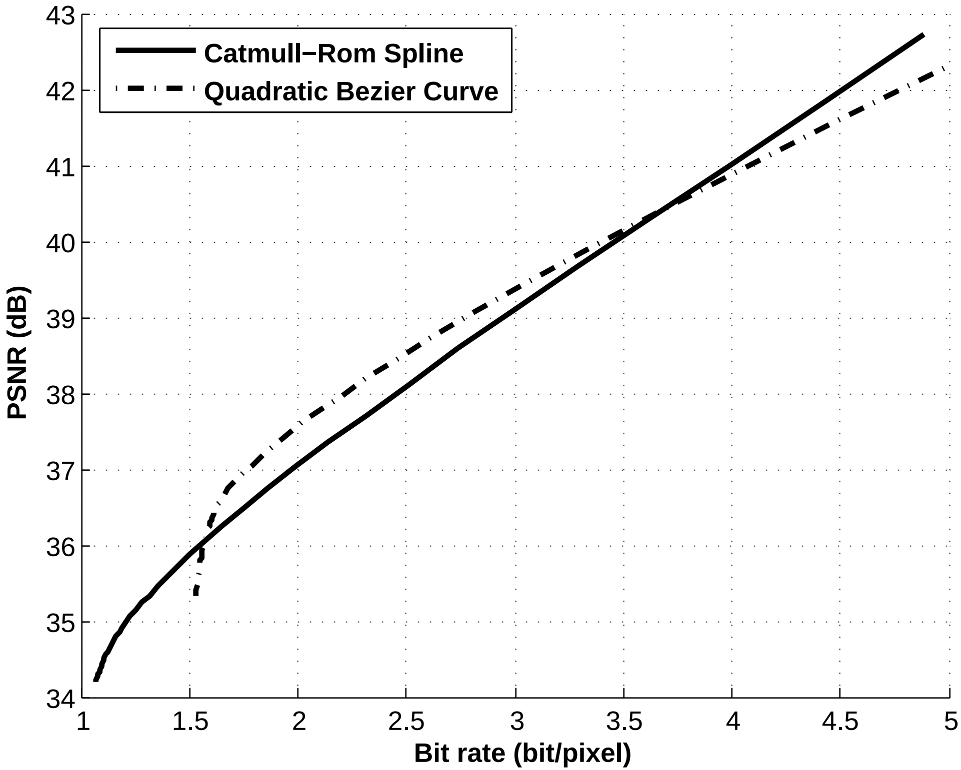
Method	*Salesman*		*Foreman*	
Name	PSNR	Bit-rate	PSNR	Bit-rate
TSS	38.132	1.7768	35.875	2.7509
CRS	38.244	0.8574	35.589	2.0687
QBC	38.291	0.8578	35.812	2.2516



Rate-distortion curves of Catmull-Rom spline & Quadratic Bézier curve, *Salesman* sequence.



Rate-distortion curves of Catmull-Rom spline & Quadratic Bézier curve, *Foreman* sequence.



Rate-distortion curves of Catmull-Rom spline & Quadratic Bézier curve, *Darius* sequence.

30th frame of *Salesman* video sequence, $\xi^{lmt} = 100$. Top: CRS approximated frame, 38.68-dB, 0.92991-bpp. Bottom: QBC approximated frame, 39.968-dB, 1.0796-bpp.

30^{th} frame of *Foreman* video sequence, $\xi^{lmt} = 100$. Top: CRS approximated frame, 37.591-dB, 2.401-bpp. Bottom: QBC approximated frame, 38.191- dB, 2.7374-bpp.

30^{th} frame of *Darius* video sequence, $\xi^{lmt} = 100$. Top: CRS approximated frame, 41.838-dB, 4.4234-bpp. Bottom: QBC approximated frame, 42.272-dB, 4.9813-bpp.

Home Page
Title Page
Contents
Page 23 of 24
Go Back
Full Screen
Close
Quit

8. Conclusion & Future Work

⋆ We presented a method of compression of temporal video data by Catmull-Rom Spline and Quadratic Bézier Curve Fitting.

⋆ Experimental results show that the proposed method yields very good results both in terms of objective and subjective quality measurement parameters, i.e. bit-rate/PSNR and human visual acceptance, without causing any blocking artifacts..

⋆ Compression of spatial video data by spline/curve fitting is under investigation.

THANKING YOU