GENERALIZED SELECTIVE DATA PRUNING FOR VIDEO SEQUENCE

Yuichi Tanaka, Madoka Hasegawa and Shigeo Kato

Graduate School of Engineering, Utsunomiya University
7-1-2 Yoto, Utsunomiya, Tochigi, 321-8585 Japan
Email: {tanaka, madoka, kato}@is.utsunomiya-u.ac.jp

ABSTRACT

In this paper, we extend a rate-dependent content-aware image retargeting method, generalized selective data pruning, to video sequence. It is regarded as a spatial-domain video retargeting, i.e., reduction of the spatial video size, with a rate-dependent property for side information. It is realized by the piecewise linear approximation of seam carving. The side information bitrate for video retargeting can vary depending on the control parameter and is negligible to transmit with the encoded video bitstream while the method maintains video retargeting performance.

Index Terms— Content-aware video retargeting, selective data pruning, video coding, H.264/AVC

1. INTRODUCTION

Content-aware image retargeting is a developing issue in computer graphics researches [1–3]. It is regarded as a sophisticated image resizing method. To resize an image into a different aspect ratio and/or one with complex structures, image retargeting gives better results than traditional scaling and cropping. However, it generally requires high computational complexity in order to analyze local and/or global image structures such as objects, backgrounds, and saliency. Therefore, image retargeting is a very challenging task for the built-in software in portable devices.

For video sequence, there are several approaches for content-aware retargeting. Two implementations, spatial domain retargeting and temporal domain retargeting are possible for video sequence. The spatial one is a straightforward extension of image retargeting which shrinks spatial size of the video. In contrast, temporal one is used to reduce the number of frames of the sequence, which is a sophisticated solution of frame dropping. In this paper, we focus on spatial domain video retargeting.

Seam carving (SC) [1, 2] is a pioneering work for image/video retargeting. At first it was defined as an image retargeting method, and then it has been extended for video retargeting. A seam in an image is defined as an eight-connected path of pixels from the top (left) to the bottom (right) of the original image. The width (height) of the image is reduced by one pixel after pruning a seam. The final retargeted image is produced with iterative pruning of the seams until the retargeted image reaches the desired width (height). For video sequences, a dynamic (temporally changing) seam is produced to cut the video sequence by a *video seam*; each coordinate in a video seam is eight-connected spatially and temporally. Hence a video seam cuts a video cube from the top (left) to the bottom (right), and from the first frame to the last one. By using a graph cut approach, the original seam carving realizes to implement a dynamic video seam [2]. Another method is possible such as discontinuous seam carving [3] to calculate more efficient seams.

However, there are few approaches to implement seams in a rate-dependent manner. Since retargeting is often required in portable devices with restricted resolution displays rather than powerful desktop computers, SC is still a computationally demanding process for such portable devices. As a straightforward approach, video seam paths by SC calculated at a server side can be transmitted to a client. Unfortunately, the bitrates of the seam paths demand a very high percentage of the entire bit budget for the transmitted video. It is a significant drawback of the straightforward application of SC.

The authors have proposed generalized selective data pruning (GenSDP) of images [4, 5], which is defined as a rate-dependent seam carving. It is inspired by SC and selective data pruning (SDP) [6], which is an image compression method with downsampling insignificant lines in an image. GenSDP enables seam carving-like implementation of downsampled paths instead of line-wise one in SDP. In this paper, we extend GenSDP to video sequences. It generates dynamic seams with rate-dependent properties and the required bitrates to represent seam paths can be negligible if the video is encoded with middle to high quality.

Notations: Table 1 represents the notation of used symbols for an image I in this paper. To simplify the description, we use Matlab-based notations for sets of rows and columns in a matrix.

Table 1. Reference Table of Symbol Notation for an Image I

Symbol	Description
$I(i, j)$	Pixel value at i-th row and j-th column
$I(i, :)$, $I(:, j)$	i-th row, j-th column
$I(L_0 : L_1, :)$	Sub-image where $L_0 \le i \le L_1$

For simplicity, a vertical seam is considered in this paper. It can be easily extended for the horizontal situation.

2. REVIEW

2.1. Seam Carving

SC [1, 2] prunes seams from an image which are classified as insignificant. Also, it can insert seams in an image to expand the image size. In the original SC for images, a dynamic programming approach is used to decide which seam can be pruned. For other details, please refer to [1, 2]. Note that seam pruning is not a reversible process. That is, when a seam was pruned, its seam path cannot be recovered from the pruned image without seam path information (SPI).

This work was supported in part by KAKENHI 22760263 and SCAT Research Grant.

2.2. Selective Data Pruning

SDP is a simple line-based image downsampling method. It is described as follows: Target columns to be pruned are somehow found from an image, and then they are pruned to reduce the width of the image. The original paper [6] uses a MSE-based metric to determine the columns to be pruned. SDP is proposed for a "decimation-then-compression" approach of image coding. The resized image by SDP is encoded by any image encoders, then the compressed image is interpolated at the receiver side with high-order edge-directed interpolation.

As reported in [6], it is possible to use SDP for video coding. Pruned positions for frames are the same within a GOP. With the high-order edge-directed interpolation, it outperforms the normal H.264/AVC in the very low bitrate case.

3. GENERALIZED SELECTIVE DATA PRUNING

GenSDP has two variations. One is based on hierarchical seam search method, and the other is used piecewise linear approximation (PLA) [4, 5]. In this paper, the PLA-based GenSDP (GenSDP-PLA) is modified to obtain dynamic seams for video sequences, thus we review its essential aspects.

3.1. Slope Information

Let $\boldsymbol{s}_{O,n}(i)$ be the i-th SPI of the n-th seam, $\boldsymbol{s}_{O,n}$, calculated by the original SC. The pruned pixel position is determined by $I(i, \boldsymbol{s}_{O,n}(i))$. The PLA of $\boldsymbol{s}_{O,n}$ is specified as a set of slopes. The slope G is defined as follows:

$$G = \arctan \Delta j / \Delta i \tag{1}$$

where Δi and Δj are respectively the vertical and horizontal distances from the starting point (in pixel coordinates), $\Delta i \geq 0$, and $\left|\frac{\Delta j}{\Delta i}\right| \leq \tan\frac{\pi}{4}$. The last condition is the requirement for a vertical seam. Let $\boldsymbol{s}_{E,n}$ be n-th estimated SPI to be pruned. For each L row of I, $\boldsymbol{s}_{E,n}$ is approximated from $\boldsymbol{s}_{O,n}$ and there is a restriction $\boldsymbol{s}_{E,n}(kL+1) = \boldsymbol{s}_{E,n}(kL)$, i.e., the last seam position of the previous L rows and the first seam position of the current L rows are the same to reduce the amount of side information. Consequently, the side information consists of a set of used Gs, the overall starting point $\boldsymbol{s}_{E,n}(1)$, and a set of Ls.

3.2. Recursive Optimization of the Estimated Seams

Assume that $\boldsymbol{s}_{O,n}$ is calculated before the estimation. Therefore, the parameters known just before the optimization are $\boldsymbol{s}_{E,n}(1) := \boldsymbol{s}_{O,n}(1)$ and $\boldsymbol{s}_{O,n}$. It is clearly understood that the optimal unit length $L_{opt} \leq L$ is different in each portion of I. To estimate $\boldsymbol{s}_{E,n}$ effectively, we define the simplified forward energy criterion shown as follows:

$$C_{\text{SFE}}(\boldsymbol{s}_n) = \frac{1}{L_{\boldsymbol{s}_n}} \sum_{i \in \text{row index of } \boldsymbol{s}_n} |I(i, \boldsymbol{s}_n(i)-1) - I(i, \boldsymbol{s}_n(i)+1)| \tag{2}$$

where $L_{\boldsymbol{s}_n}$ is the length of $\boldsymbol{s}_n$. This criterion is similar to the forward energy criterion of the improved SC [2] but nevertheless is less complex. In the optimization process, we use (2) since many repetitive calculations are required.

Algorithm 1, shown below, presents the detailed optimization process of $\boldsymbol{s}_{E,n}$ for $I(L_0 : L_0 + L - 1, :)$. In Algorithm 1, $\boldsymbol{G} = \{G_0, G_1, \ldots, G_{N_G-1}\}$, where N_G is the number of slope candidates, DIRECQUANT$(a_0, \boldsymbol{a})$ is a function that finds the nearest value to a_0 from a set $\boldsymbol{a}$, and $f_{\text{PLA}}(j, G, l)$ returns an integer lattice, formalized as follows:

$$f_{\text{PLA}}(j, G, l) = \{x \,|\, x = \lfloor Gy + 0.5 \rfloor + j \wedge 0 \leq y \leq l-1\} \tag{3}$$

where $\lfloor \cdot \rfloor$ is a floor operator, and $T_{\text{PLA}} > 0$ is a threshold. It is clear that $\boldsymbol{s}_{E,n}$ can be completely recovered from the returned parameters: the PLA unit length L, the terminated coordinate $\boldsymbol{s}_{E,n}(L_0 + L - 1)$, which is equal to $\boldsymbol{s}_{E,n}(L_0 + L)$, and the optimal slope G_{opt}.

Algorithm 1 Recursive Seam Estimation with PLA

function SEAMESTPLA$(I(L_0 : L_0 + L - 1, :)$, $\boldsymbol{s}_{E,n}(L_0)$, $\boldsymbol{s}_{O,n}(L_0 : L_0 + L - 1)$, $\boldsymbol{G}$, $T_{\text{PLA}})$
  $G_{opt} \leftarrow$ DIRECQUANT$\left(\frac{\boldsymbol{s}_{O,n}(L_0+L-1) - \boldsymbol{s}_{E,n}(L_0)}{L-1}, \boldsymbol{G}\right)$
  $\boldsymbol{s}_{E,n}(L_0 : L_0 + L - 1) \leftarrow f_{\text{PLA}}(\boldsymbol{s}_{E,n}(L_0), G_{opt}, L)$
  $J_{orig} = C_{\text{SFE}}(\boldsymbol{s}_{O,n}(L_0 : L_0 + L - 1))$
  $J_{est} \leftarrow C_{\text{SFE}}(\boldsymbol{s}_{E,n}(L_0 : L_0 + L - 1))$
  if $(J_{orig} - J_{est}) > T_{\text{PLA}}$ **and** $L > 2$ **then**
    SEAMESTPLA$(I(L_0 : L_0 + L/2 - 1, :)$, $\boldsymbol{s}_{E,n}(L_0)$, $\boldsymbol{s}_{O,n}(L_0 : L_0 + L/2 - 1)$, $\boldsymbol{G}$, $T_{\text{PLA}})$
    SEAMESTPLA$(I(L_0 + L/2 : L_0 + L - 1, :)$, $\boldsymbol{s}_{E,n}(L_0 + L/2 - 1)$, $\boldsymbol{s}_{O,n}(L_0 + L/2 : L_0 + L - 1)$, $\boldsymbol{G}$, $T_{\text{PLA}})$
  else
    return L, $\boldsymbol{s}_{E,n}(L_0 + L - 1)$, G_{opt}
  end if
end function

3.3. Iterative Optimization

The rate-dependent seams are optimized by using a cost function with a Lagrange multiplier. The seam estimation is realized by an iterative optimization of the recursive estimation introduced above. The object of GenSDP-PLA is to find the nearest seam $\boldsymbol{s}_{E,n}$ to $\boldsymbol{s}_{O,n}$ with a bitrate constraint for SPI, which is represented as follows:

$$\begin{aligned} \boldsymbol{s}_{E,n} &= \arg\min_{\boldsymbol{s}_C}(J_{\text{PLA}}(\boldsymbol{s}_C)) \\ J_{\text{PLA}}(\boldsymbol{s}_C) &= D_{\text{PLA}}(\boldsymbol{s}_C) + \lambda R_{\text{PLA}}(\boldsymbol{s}_C) \end{aligned}. \tag{4}$$

In [4], $D_{\text{PLA}}(\boldsymbol{s}_C)$ is defined based on the reconstruction error after interpolation back of the $\boldsymbol{s}_{E,n}$ by the high-order edge-directed interpolation. The process is iterated for all possible T_{PLA}s in Algorithm 1 until the optimum T_{PLA} is found for a given λ. Details are described in [4].

3.4. Bitrate Calculation

In this paper, we calculate bitrates for the unit length set with a simple binary representation. In Algorithm 1, `0` is stored if the seam is not divided, whereas `1` is stored if the seam is divided. After the recursive optimization, we obtain a binary vector $\boldsymbol{b}_{\text{div}}$ for the division information of the candidate $\boldsymbol{s}_C$. Finally, $R_{\text{PLA}}(\boldsymbol{s}_C)$ can be represented as follows:

$$R_{\text{PLA}}(\boldsymbol{s}_C) = \mathcal{N}(\boldsymbol{b}_{\text{div}}) + \mathcal{N}(\boldsymbol{b}_{\text{div},0}) \lceil \log_2 N_G \rceil \tag{5}$$

where $\mathcal{N}(\boldsymbol{b}_{\text{div}})$ is the number of elements in $\boldsymbol{b}_{\text{div}}$, $\boldsymbol{b}_{\text{div},0}$ is the number of `0`s in $\boldsymbol{b}_{\text{div}}$. The term $\mathcal{N}(\boldsymbol{b}_{\text{div},0}) \lceil \log_2 N_G \rceil$ is reserved for the slope information.

4. GENSDP EXTENSION FOR VIDEO SEQUENCE

In this section, we introduce the extension of GenSDP for video sequences based on the method described above. GenSDP-PLA is used to calculate initial seams in the first frame and they are used to seek dynamic seams for the following frames. In the following expressions, we omit some subscript for simplicity. For example, $\boldsymbol{s}_{E,n}(i) \rightarrow \boldsymbol{s}_E(i)$.

4.1. Calculation of Initial Seams

In the original SC for video [2], a graph cut method is used since dynamic programming used in image SC cannot be applied to construct a video seam. However, it is also difficult to include above-mentioned rate-dependent characteristics into the graph cut method. Therefore, video seam calculation based on the initial image seam is presented.

In the proposed approach, the initial seams are calculated and pruned from the first frame. The dynamic seams for remaining frames are sequentially calculated from that of the previous frame. The reference frame R for the initial seam calculation is defined as follows:

$$R = I_0 + \alpha \sum_{\hat{t}=1}^{N_{GOP}} |I_{\hat{t}} - I_{\hat{t}-1}| \tag{6}$$

where I_t $(t = 0, \ldots, N_{GOP})$ is the t-th frame of the video sequence, and N_{GOP} is the GOP length for the initial seam calculation. For this reference frame, rate-dependent seams are calculated by using GenSDP-PLA.

4.2. Calculation of Dynamic Seams

From the initial seam, dynamic seams are calculated frame by frame. The seams for $t \geq 1$ have a necessary condition represented as follows:

Condition 1 (pixel connectivity)

$$\left|\boldsymbol{s}_E^{t_C}(i_C) - \boldsymbol{s}_E^t(i)\right| \leq 1 \tag{7}$$

where $\boldsymbol{s}_E^t(i)$ is an estimated seam for t-th frame and $t_C = \{t-1, t\}$ and $i_C = \{i-1, i, i+1\}$.

This is the strict requirement from the connectivity condition of a video seam. Furthermore, we append a couple of extra conditions for the dynamic seam calculation to reduce side information:

Condition 2 (seam division)

$$\boldsymbol{b}_{div}^t = \boldsymbol{b}_{div}^0 \quad for\ \forall t \tag{8}$$

The condition means the division information at $t = 0$ is shared with all frames.

Condition 3 (pixel connection between divisions) *Similar to GenSDP for images, $\boldsymbol{s}_E^t(kL+1) = \boldsymbol{s}_E^t(kL)$ for $\forall t, k$.*

It simplifies the dynamic seam calculation since for each division of I, i.e., $I((k-1)L+1 : kL, :)$, the starting point $\boldsymbol{s}_{E,n}((k-1)L+1)$ is automatically set.

From these conditions, the dynamic seam search for $\boldsymbol{s}_E^t((k-1)L+1 : kL)$ is greatly simplified as follows:

1. Set $\boldsymbol{s}_E^t((k-1)L+1)$ by $\boldsymbol{s}_E^t((k-1)L)$.

2. Restore three seam candidates as follows:

$$\boldsymbol{s}_{C,u}^t = f_{\text{PLA}}(\boldsymbol{s}_E^t((k-1)L+1), G_u^{t,k}, L) \tag{9}$$

where $u = -1, 0, 1$ and

$$G_u^{t,k} = (\boldsymbol{s}_E^{t-1}(kL) - \boldsymbol{s}_E^t((k-1)L+1) + u)/L. \tag{10}$$

3. Determine the optimal seam $\boldsymbol{s}_E^t((k-1)L+1 : kL)$ with the lowest distortion shown below:

$$\boldsymbol{s}_E^t((k-1)L+1 : kL) = \arg\min_{\boldsymbol{s}_{C,u}^t}(C_{\text{CFE}}(\boldsymbol{s}_{C,u}^t)). \tag{11}$$

Fig. 1. Tenth frames of resized video. From left to right, top to bottom: original, bicubic scaling, SDP, and GenSDP. Left: *Akiyo*. Right: *Crew*.

5. EXPERIMENTAL RESULTS

In this section, some comparisons with SDP are shown. In this paper, 30 frames of popular video sequences, *Akiyo*, *Mobile*, *Crew*, and *Football* with CIF size (352 $\times$ 288) are used for the experiment. Only vertical seams/lines are calculated and pruned for simple discussions. The number of removed seams/lines are fixed to 88, which is a quarter of the column size. We set control parameters $\lambda = 10, 100$ in (4).

5.1. Resized Video Quality

The first comparison is the retargeting performance. Fig. 1 is the comparison of the tenth frame after bicubic scaling, SDP, and GenSDP. Only *Akiyo* and *Crew* are shown due to limitation of space, but all retargeted videos have similar characteristics described below. The scaled frames are shown for comparison purpose. Clearly scaling does not work well if aspect ratio of the videos is changed. SDP presents better resizing performance than scaling, however, some important regions are discarded since the positions of the pruned lines are the same for the whole GOP. In contrast, our approach presents the best retargeted frames among the three due to the realization of dynamic seams. The estimated seams by GenSDP are shown in Fig. 2. It shows the initial frame ($t = 0$) and the 29th frame. It is clear that our rate-dependent dynamic seams avoid to cross salient regions.

To measure the retargeting performances, we interpolate the resized frames back to the original resolution, then measure PSNRs. It is a possible method to check how better a data pruning method keeps the original ROIs. We applied high-order edge-directed interpolation [6] for each reduced size frames. Table 2 summarizes the interpolation performance. We omit PSNR of GenSDP where $\lambda = 100$ for *Akiyo* since it is the same as the case of $\lambda = 10$. GenSDP gains significant PSNR improvements from SDP. Especially for *Akiyo*, the proposed method shows >5 dB better PSNR than SDP.

Fig. 2. Estimated dynamic seams of *Crew*. Left: initial frame. Right: 29th frame.

Table 2. Interpolation Back PSNRs (dB)

	SDP	GenSDP	
		$\lambda = 10$	$\lambda = 100$
Akiyo	43.33	48.96	-
Mobile	26.81	29.46	29.22
Crew	42.93	43.17	43.02
Football	35.09	35.60	35.14

It is worth noting that GenSDP can be directly applicable to 2-D retargeting by using its algorithms for the columns and rows. For the optimal sequence of reducing row/column size, we can use a dynamic programming approach similar to that of [1].

5.2. Video Coding Performance

The final comparison is for video coding performance. In the application for video coding using GenSDP, we have two variations that transmit:

Case 1 Compressed full-size video and side information.

Case 2 Compressed reduced-size video and side information.

We consider Case 1 in this paper since in our experiment, Case 1 performs better for almost overall quality parameters for H.264 due to its powerful intra/inter picture prediction. The R-D curves are illustrated in Fig. 3. The figure also contains the curve by H.264 with SDP. For SDP, we show only Case 2 since it is not designed for video retargeting.

For videos with very large static regions, such as *Akiyo*, SDP is comparable to H.264 alone in very low quality encoding. However, the other video sequences and the high bitrate case of *Akiyo*, H.264 performs better than SDP. Our approach in Case 1 shows very comparable R-D curves to H.264 alone, since the bitrate for side information is negligible for that case.

It is clear that our proposed method does not lose H.264's compression performance even if the side information is transmitted with the H.264 bitstream. In our experiments, seam information requires only 20-30 kbps with $\lambda = 100$ and 20-190 kbps with $\lambda = 10$ for video sequences. The videos with complex motions, such as *Mobile* and *Football* require relatively many bits to represent estimated dynamic seams. However, such videos also need high bitrates for textures and motion vectors. As a result, the ratio of side information for seams to be pruned requires a small part of the entire bit budget. Therefore, it is a good choice to encode videos always with Case 1 if we need a retargeted video as well as full-resolution one.

In comparison with SC for side information, SC requires much more bitrates than SDP and GenSDP. For example, SC demands almost 3.9×10^4 bits (~ 37 kbps) if 88 seams are pruned from a CIF size *image* (not video sequence). For video sequences, the required bitrates raise proportionally depending on the number of frames.

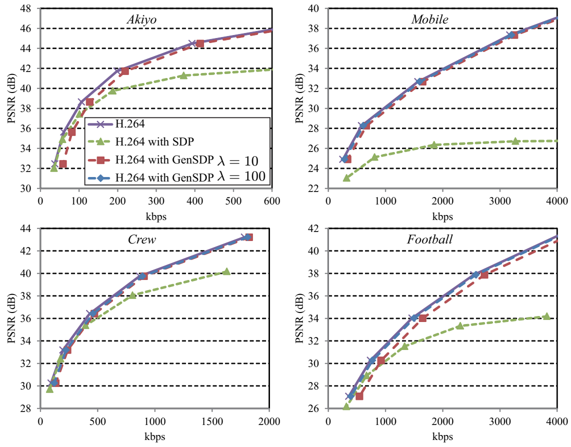

Fig. 3. R-D curves for video coding. From left to right and top to bottom: *Akiyo*, *Mobile*, *Crew*, and *Football*. We omit the curve of GenSDP where $\lambda = 100$ for *Akiyo* since it is the same as the case of $\lambda = 10$.

6. CONCLUSIONS

In this paper, we propose a rate-dependent video retargeting method as an extension of GenSDP into spatial-temporal domain. It is realized by the sequential calculation of dynamic seams from the initial spatial seam, which is obtained from GenSDP for images. The retargeted video maintains salient regions in the original videos, whereas the required bitrates for side information can be negligible for high bitrate video encoding. There could be more efficient/fast extensions for GenSDP. For example, rate-dependent properties will be incorporated with graph cut of videos to obtain globally-optimum video seams. Furthermore, we can refresh the position of seams for a frame by using the discontinuous seam carving approach [3]. Our future work includes to describe a better way to implement rate-dependent video seams.

7. REFERENCES

[1] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graph.*, vol. 26, no. 3, 2007.

[2] M. Rubinstein et al., "Improved seam carving for video retargeting," *ACM Trans. Graph.*, vol. 27, no. 3, 2008.

[3] M. Grundmann et al., "Discontinuous seam-carving for video retargeting," in *Proc. IEEE CVPR*, 2010, pp. 569–576.

[4] Y. Tanaka et al., "Seam carving with rate-dependent seam path information," in *Proc. ICASSP'11*, 2011, to be presented.

[5] Y. Tanaka et al., "Improved image concentration for artifact-free image dilution and its application to image coding," in *Proc. ICIP'10*, 2010, pp. 1225–1228.

[6] D. T. Võ et al., "Selective data pruning-based compression using high-order edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 399–409, 2010.