TABLE OF CONTENTS

ABSTRACT 1

1 Introduction 2

1.1 Motivation 2

1.2 Related works 4

1.3 Dissertation overview 5

2 Basic formulation of video synopsis 5

2.1 Activity energy 6

2.2 Time-lapse background generation 10

2.3 Background consistency energy 10

2.4 Temporal consistency energy 11

2.5 Collision energy 13

2.6 Computational bottleneck 14

3 Proposed tube rearrangement 15

3.1 Occupation matrix generation 15
3.2 Objective function 20
3.3 Optimizing objective function 22
3.4 Parallelized optimization 25
4 Online video synopsis framework 27
4.1 Background modeling 28
4.2 Object tube generation 28
4.3 Object stitching 28
5 Experimental Results 29
5.1 Efficiency of parallelized tube rearrangement 31
5.2 Comparisons of different online tube rearrangement algorithms 31
6 Conclusion 32
BIBLIOGRAPHY 40

LIST OF FIGURES

1 Concept diagram of video synopsis [1]. The bird and man appeared at different time in the original video are rearranged in temporal domain, and then displayed simultaneously in the condensed video. 7

2 Example of the object tube rearrangement in 2D space. Red arrows indicate some offsets of the starting labels for better understanding of the tube rearrangement process. We can see that after the tube rearrangement, the length of the condensed video becomes much shorter than that of the original. . . . 8

3 Flowchart of the proposed online video synopsis framework. At the beginning, foreground of the object tube is reshaped into the 3D occupation matrix. We use this matrix representation to calculate the collision energy fast in conjunction with parallel processing and Fourier transform. Afterwards, we determine optimal starting labels for tubes and stitch tubes with the background to generate a resulting synopsis video. Illustrations of man and vehicle in this figure are created by Lluisa Iborra and Yasser Megahed from the Noun Project. . . 16

4 Example of the binary occupation matrix generation when $W/N = 16$ and $H/M = 9$. The foreground and background of the object are represented in black and white, respectively. Dotted lines in the figure are depicted to show contours of the original object for the readers. Illustrations of the woman and man in this figure are created by Nataliia Lytvyn and Ludovic Gicqueau from the Noun Project, respectively. 19

5 Example calculation of reformulated collision energy E_c with two binary occupation matrices. Occupied elements in the matrix are colored in red and blue. After the element-wise multiplication, we can see that the objects have two collided elements colored in magenta. In this figure, $\odot$ is an operator for the element-wise multiplication, also known as Hadamard product []. 21

6 Two ways of calculating E_c. All of occupation matrices in this figure have $6 \times 8 \times 3$ spatio-temporal resolution. E_c can be calculated by using (a) Hadamard products between two sets of frames, and (b) 1D cross correlations between 48 pairs of 1D signals. Three primitive colors (red, green, and blue) in this figure is used to show example correspondences. 26

7 Examples of the test sequences. All sequences were captured at Hanyang university, Seoul, Korea. 30

8 Running times of the different tube rearrangement algorithms for six test sequences. An asterisk (*) indicates the parallelized version of the proposed algorithm. 32

9 Frame condensation ratios of different tube rearrangement algorithms for six test sequences. 33

10 Compact ratios of the different tube rearrangement algorithms for six test sequences. 34

11 Overlap ratios of the different tube rearrangement algorithms for six test sequences. 34

LIST OF TABLES

1 List of test sequences used in the experiments 29

ABSTRACT

Video synopsis allows us to analyze security videos efficiently by condensing or shortening a long video into a short one. To generate a condensed video, moving objects (a.k.a. object tubes) in the video are rearranged in the temporal domain using a predefined objective function. The objective function consists of several energy terms which play important roles in making a visually appealing condensed video. One of the energy terms, collision energy, creates a bottleneck in the computation because it requires two object tubes to calculate the degree of collision between them. Existing approaches try to reduce the computation time of the collision energy calculation by reducing the number of tubes processed at once. However, those approaches are not sufficient to generate condensed video when the number of object tubes becomes large.

In this letter, we propose a fast Fourier transform (FFT)-based parallelized tube rearrangement algorithm. To take advantage of both parallel processing and FFT, we represent object tubes as 3D binary matrices (occupation matrices). An objective function of the tube rearrangement problem is defined on the occupation matrix, and a starting position for each tube in the temporal domain is then determined by optimizing the objective function. Throughout the experiments, the proposed algorithm took a much shorter time to condense the video than existing algorithms, while other performance metrics were similar.

1 Introduction

1.1 Motivation

The field of security video summarization has been studied for decades to reduce burdens of browsing large amount of video footages. Earlier approaches [2–4] prior to video synopsis [5–7] suffered from several disadvantages including low frame condensation ratio (FR) or missing information, when the frame length of the input video was long. Fundamental building blocks of such approaches were image frames, which means that they tried to select a subset of image frames representing the original video best. On the other hand, building blocks of video synopsis [5–7] are moving objects extracted from the scene, called *object tubes*. In the video synopsis framework, the object tubes are rearranged in the temporal domain and stitched back with background images to generate a short and condensed video. This difference allows video synopsis to efficiently utilize the spatial domain of the video and to drastically improve the FR as compared to the earlier approaches.

Among the diverse research topics in video synopsis, solving the optimization problem for determining starting positions (starting labels) of the object tubes in the temporal domain greatly affects the system performance regarding computation time. This problem is simply denoted as *a tube rearrangement problem*.

In the pioneering work of video synopsis by Pritch *et al.* [7], the tube rearrangement problem is formulated as Markov Random Fields (MRFs) [8] with four energy terms: activity, collision, temporal consistency, and background consistency. The starting label for each

object tube is then determined by minimizing the energy function of MRFs with a simulated annealing [9] or greedy optimization algorithm [10]. During the optimization process, calculating pairwise energy terms (in this case, collision and temporal consistency) becomes a bottleneck for computation speed, because such calculation has $O(TK^2)$ complexity, where T is the number of time steps and K is the total number of object tubes.

In order to cope with the problem, Pritch *et al.* [1] suggest a clustering based optimization algorithm. It divides object tubes into several subsets; then, the optimization algorithm is conducted on each subset. Since the number of object tubes belonging to each subset is much smaller than K, execution time of the optimization algorithm is greatly reduced. However, its condensation result depends on the performance of the clustering algorithm which has a chance to generate inappropriate clusters.

An alternative approach to tube rearrangement is an online video synopsis [11–15], which solves a stepwise optimization problem. In stepwise optimization, instead of considering entire object tubes at the same time, the starting labels of the object tubes are determined one by one. Therefore, it requires less computational power and memory space than batch or offline video synopsis. In addition, since online video synopsis optimizes the object tubes in chronological order, it is inessential to consider temporal and background consistencies. Therefore, the most of online video synopsis frameworks mainly consider the collision energy during the optimization.

Based on such advantages, recent studies of online video synopsis focused on finding efficient ways of solving a stepwise optimization problem: for example, the maximum a posteriori estimation [11], a Tetris-like tube rearrangement strategy [12, 13], and a potential collision graph [14]. Even though these existing algorithms have their own virtue, they have

lack of considerations of multi-core environment, which means that there are still rooms for improvements.

In this dissertation, the tube rearrangement problem is reformulated as a suitable form for parallel processing, and the novel concurrent optimization algorithm based on 1D convolutions is proposed to accelerate the optimization process. As in other online tube rearrangement algorithms [11–15], the collision energy is primarily considered during the tube rearrangement. As a preprocessing step, the proposed algorithm reshapes object tubes into probabilistic occupation matrices of $\mathcal{M} \times \mathcal{N} \times \mathcal{T}$ dimension, where $\mathcal{M}$ and $\mathcal{N}$ represent the spatial domain, and $\mathcal{T}$ represents the time domain. This occupation matrix becomes a fundamental building block of the proposed algorithm. Then, the collision energy between two object tubes can be computed as element-wise multiplications of two occupation matrices. This process can be accelerated by utilizing Fast Fourier transform (FFT) [16] in conjunction with parallel computing; therefore, the proposed algorithm can effectively determine the starting labels of numerous object tubes in very short amount of time.

1.2 Related works

A summary of the recent advances in video synopsis is presented as follows. Nie *et al.* [17] rearrange object tubes in both temporal and spatial domain to generate more condensed videos. Zhu *et al.* [18] and Mahapatra *et al.* [19] extend the concept of video synopsis to the multi-camera network. Wang *et al.* [20] and Zhong *et al.* [21] utilize the compressed domain to generate synopsis videos efficiently. X. Li *et al.* [22] scale down object sizes to reduce collisions in the synopsis video. Z. Li *et al.* [23] and K. Li *et al.* [24] introduce a seam

carving method to remove redundant information from the original video.

1.3 Dissertation overview

The rest of the dissertation is organized as follows. Section 2 introduces the basic formulation of video synopsis and details of the proposed tube rearrangement algorithm are described in Section 3. Section 4 contains explanations of other components that the online video synopsis consists of. Section 5 presents experimental results, and the dissertation is concluded in Section 6.

2 Basic formulation of video synopsis

In this section, the basic formulation of video synopsis introduced in the pioneering works [5–7] is described to show which part of the formulation has to be changed for parallel processing. In addition, the reason why online video synopsis mainly considers collision energy is explained in detail.

As in Fig. 1 and Fig. 2, a principal objective of video synopsis is shortening length of the input video by relocating object tubes in temporal domain. In other words, we try to find the best combination of object tubes' starting positions in temporal domain (starting labels). In the field of video surveillance, a definition of the best combination can be different from specific applications. However, based on the paper of Pritch *et al.* [7], the condensed video with the best starting label combination should have following characteristics.

- Objects of interests should be appeared in the condensed video.

- Rearranged object tubes should seamlessly rendered in the condensed video.
- The condensed video has significantly shorter length than the input video.
- Dynamics of objects or interactions between the objects should be understood in the condensed video.

To achieve the characteristics, the batch video synopsis [7] utilizes four energy terms as described in Section 1: activity, background consistency, collision, and temporal consistency. The order of the energy terms are matched with that of the characteristics.

Assume that $L = \{l_0, ..., l_N\}$ is a set of starting labels for N object tubes; then, an objective function $E(L)$ can be defined as

$$E(L) = \sum_{l_i \in L} \left(E_a(l_i) + \gamma E_s(l_i) \right) + \sum_{l_i, l_j \in L} \left(\alpha E_t(l_i, l_j) + \beta E_c(l_i, l_j) \right), \tag{1}$$

where E_a, E_s, E_t, and E_c are activity, background consistency, temporal consistency, and collision energies, respectively. In addition, α, β, and γ are weighting parameters for controlling importance between the energies.

2.1 Activity energy

At first, E_a defines which object tubes should be appeared in the condensed video. One example of E_a is

$$E_a(l_i) = \begin{cases} \sum_{x,y,t} \chi_i(x, y, t) & l_i \in L_e \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

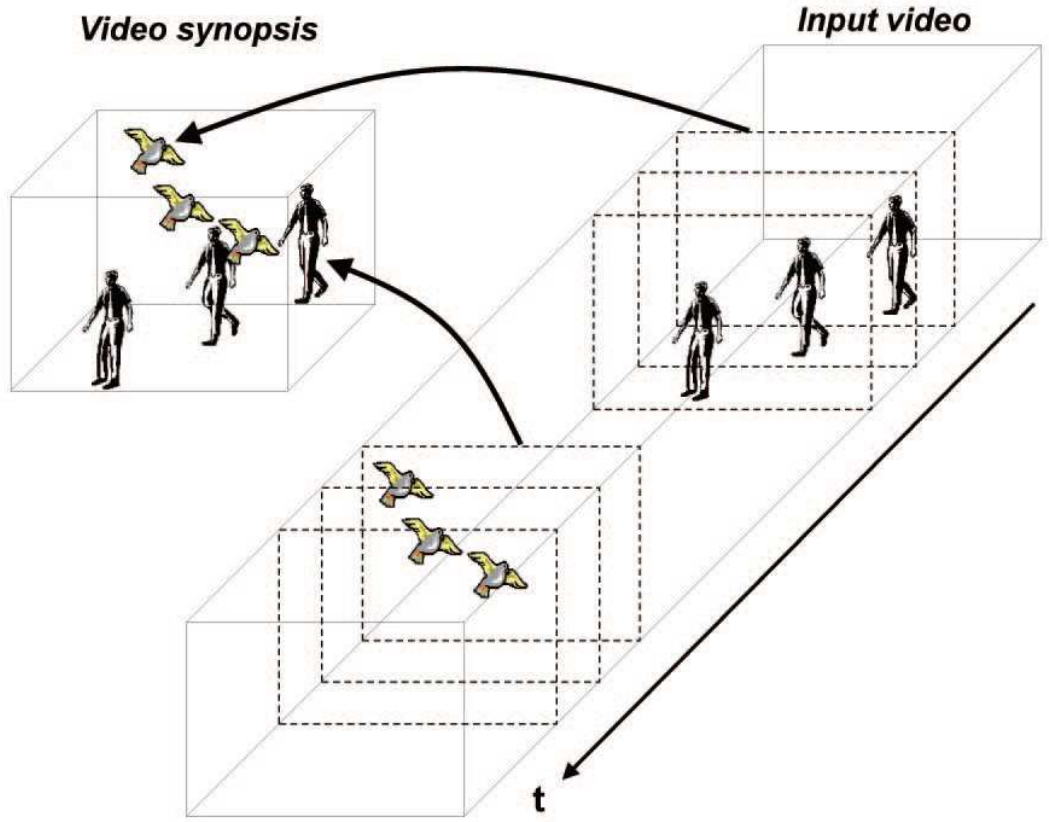


Fig. 1. Concept diagram of video synopsis [1]. The bird and man appeared at different time in the original video are rearranged in temporal domain, and then displayed simultaneously in the condensed video.

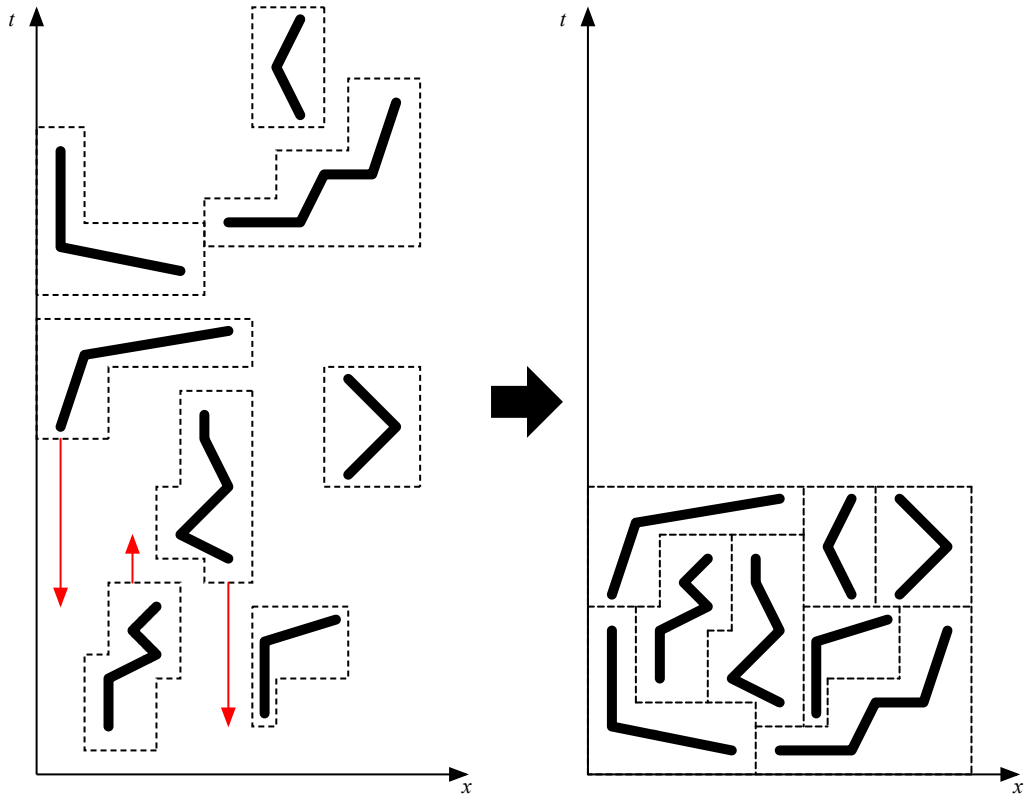


Fig. 2. Example of the object tube rearrangement in 2D space. Red arrows indicate some offsets of the starting labels for better understanding of the tube rearrangement process. We can see that after the tube rearrangement, the length of the condensed video becomes much shorter than that of the original.

where l_i and $\chi_i(x, y, t)$ are the starting label and the characteristic function of the i^{th} object tube, respectively. Due to the condition ($l_i \in L_e$) in (2), the only characteristic function of the object tube whose starting label belongs to L_e is added to E_a. The set L_e contains starting labels of the objects not included in the condensed video. Therefore, the role of E_a is penalizing exclusions of the object tubes. On the other hand, $\chi(x, y, t)$ represents the importance of the object tube. If the characteristic function of one object has larger values than that of the others, the object is more likely to be included in the resulting video. In the original work of video synopsis [5–7], $\chi_i(x, y, t)$ is defined as

$$\chi_i(x, y, t) = \begin{cases} \left\| I_i(x, y, t) - B(x, y, t) \right\| & t \in t_i \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where $I_i(x, y, t)$ is a foreground pixel of i^{th} object and $B(x, y, t)$ is a respective background pixel, and t_i is a period of time in frames indicating the appearance of the object. Based on (3), the condensed video prefers the object tubes having distinctive colors as compared with the background.

Defining a proper E_a is important for processing the query of the video synopsis users, since it determines which objects will be included in the resulting video. However, we do not have to directly optimize E_a because object filtering step prior to the optimization with specific conditions (e.g., colors, trajectories, object types, and etc) can do the same functionality.

2.2 Time-lapse background generation

Before moving on to the next energy term, how to generate time-lapse background is briefly explained. Since the main objective of video synopsis is condensing the contents of the original video, background information as well as foreground has be condensed too. If the input video is 12 hours long and the condensed video is 10 minutes long, time-lapse background can be generated by uniformly subsampling every 720th of original background images or we can use the adaptive sampling rate proportional to (or inverse proportional to) the number of objects in the current frame []. An example of the time-lapse background generation is illustrated in Fig. **??**.

2.3 Background consistency energy

The role of the second energy term in (1), E_s, is to seamlessly render the object tubes with the time-lapse background images. In the video synopsis framework, foreground pixels of the object tubes are stitched with the background images to generate the condensed video. During the stitching process, image blending algorithms (e.g., Poisson image editing []) can be used to smoothly blend the foreground and background pixels. However, inaccurate segmentation results of the foreground or foreground and background pixels from different time of day can cause visually unappealing results as shown in Fig. **??**. E_s is defined to penalize such situation.

$$E_s(l_i) = \sum_{x,y \in \sigma_i, t} \left\| I_i(x,y,t) - B_t(x,y,t) \right\|, \tag{4}$$

where σ_i is a set of boundary pixels for the i^{th} object and $B_t(x, y, t)$ is a pixel of the time-lapse background. To obtain σ_i, we can apply morphological dilation to the foreground mask of the i^{th} object and subtract it from the original. Based on (4), the object appeared in the midnight are more likely to be appeared at night-part of the time-lapse background.

In the online video synopsis framework, object tube extraction, time-lapse background generation, and foreground-background stitching are conducted in real-time; therefore, foreground and background pixels are from the similar time of day. Therefore, online video synopsis has less reason to consider E_s during the optimization.

2.4 Temporal consistency energy

The temporal consistency energy, E_t, is designed to keep chronological orders between the object tubes in the original video. If the condensed video contains chronological disorders between the tubes, we may miss the important interaction between the objects presented in the original video. Prior to further discussion about E_t, we need to define a probability of the interaction between the two object tubes first. If the objects share common time periods in the original video ($t_i \cap t_j \neq \emptyset$), the probability becomes

$$p_I(i, j) = \exp\left(-\min_{t \in t_i \cap t_j} \frac{d(i, j, t)}{\sigma_s}\right), \tag{5}$$

where $d(i, j, t)$ is a Euclidean distance between the closest pixels of i^{th} and j^{th} objects in frame t and σ_s is a parameter for adjusting a spatial range of the interaction. Based on (5), a pair of the objects spatially adjacent to each other is more likely to have interactions between them.

On the other hand, if the objects do not have any overlap in the temporal domain of the original video, $p_I(i,j)$ is defined as

$$p_I(i,j) = \exp\left(-\frac{l_j - (l_i + T_i)}{\sigma_t}\right), \tag{6}$$

where T_i is the number of frames in the i^{th} object tube and σ_t determines a temporal proximity between the objects. In addition, (6) is defined on the assumption that the i^{th} object appears earlier than the j^{th} object in the input video ($l_i + T_i < l_j$). Therefore, the object tubes located far from each other in temporal domain are less likely to have interactions.

In summary, (5) and (6) encode the idea that objects close in spatio-temporal domain have strong interactions. Based on the two equations, we can define E_t to keep chronological orders between the objects when generating the condensed video.

$$E_t(i,j) = p_I(i,j) \cdot \begin{cases} 0 & \hat{l}_i - \hat{l}_j = l_i - l_j \\ C & \text{otherwise,} \end{cases} \tag{7}$$

where $\hat{l}$ indicates a starting label of the object in the input video and C is a large constant value to penalize the objects having temporal inconsistencies.

Since the behavior of the equation (7) is not straightforward, detail explanations will be given through examples. Assume that two objects are close in spatio-temporal domain of the original video. In this case, E_t of two objects becomes very large (due to C), when their relative starting label in the condensed video ($l_i - l_j$) is not exactly same as in the input video ($\hat{l}_i - \hat{l}_j$). Conversely, the objects far from each other in spatio-temporal domain have a

low penalty for violating the condition ($\hat{l}_i - \hat{l}_j = l_i - l_j$), because their p_I has a small value.

As similar to E_s, the role of E_t is not significant in online video synopsis. Recent online video synopsis frameworks [] maintain a queue of object tubes and the queue grows as new object tube is extracted in the input video. When the size of the queue exceeds a certain threshold K, framework generates a partial condensed video with K object tubes, and then removes the first K objects from the queue. Based on the framework, chronological disorders only can be presented when the objects are in the same part of the condensed video. Even if the objects are optimized together to generate a same part of the resulting synopsis video, their temporal inconsistencies are negligible, because their relative spatio-temporal distance is small. In consequence, the one and only energy term to optimize in online video synopsis is the collision energy.

2.5 Collision energy

The key role of E_c is to prevent the resulting synopsis video from becoming crowded. During the video synopsis process, the objects from different time periods in the input video are displayed simultaneously in the same scene of the condensed video. In this case, pixel overlaps between the objects make us difficult to understand the context of the synopsis video. To penalize such situation through E_c, a degree of collision between the objects is defined as

$$E_c(l_i, l_j) = \sum_{x,y,t \in t_i \cap t_j} \chi_i(x, y, t)\chi_j(x, y, t). \tag{8}$$

Based on (8), a collision between two objects having distinctive colors from the back-

ground is considered more seriously. However, this definition of E_c is computationally expensive due to $\chi(x, y, t)$. Therefore, in this dissertation, the multiplication of two characteristic functions is replaced with the intersection over union (IU) between two bounding boxes of the objects.

$$E_c(l_i, l_j) = \sum_{x,y,t \in t_i \cap t_j} IU\left(B_i(t), B_j(t)\right), \tag{9}$$

$$IU(B_i, B_j) = \frac{B_i \cap B_j}{B_i \cup B_j}, \tag{10}$$

where $B_i(t)$ and $B_j(t)$ are bounding boxes of i^{th} and j^{th} objects at frame t, respectively. Since the bounding box does not represent an exact location of the object, (9) can be thought as an approximated version of (8).

2.6 Computational bottleneck

In (1), we should note that energies can be categorized into two groups regarding the number of required parameters: unary and pairwise. Activity and background consistency only requires a single object tube to calculate the energies; on the other hand, remaining energies require two object tubes for the calculation. When the number of objects to optimize increases, pairwise energy terms become a bottleneck of the computation. Since E_s is not the main concern of online video synopsis, E_c becomes the one and only issue for the computational burden. As described in Section 1.2, recent studies of video synopsis **??** utilize different calculations of E_c, but their definitions of the collision energy are not suitable for parallel processing. In the following section, a new representation of the object tube named

as an occupation matrix which has a suitable form for concurrent computation of E_c will be introduced.

3 Proposed tube rearrangement

In this section, the definition of E_c is reformulated using the occupation matrix and an efficient tube rearrangement algorithm for optimizing the objective function is proposed. In addition, two types of the occupation matrix (binary and probabilistic) are introduced and their characteristics are explained in detail. A flowchart of the proposed online video synopsis framework including the tube rearrangement algorithm is illustrated in Fig. 3.

3.1 Occupation matrix generation

Each element of the occupation matrix $\mathbf{M}_i(u, v, t)$ is either from Boolean or continuous domain, and represents the probability of existence for i^{th} object tube at position (u, v) and time t of a video whose spatial resolution is $H \times W$. The i^{th} occupation matrix $\mathbf{M}_i$ is then formed by stacking resized binary foreground masks of the object over multiple frames. The resized foreground mask has $M \times N$ resolution, where M and N have much smaller values than the width and height of the original video ($M \ll H$ and $N \ll W$). In this dissertation, two strategies of resizing will be introduced in following subsections and they determine the type of resulting occupation matrix: binary and probabilistic.

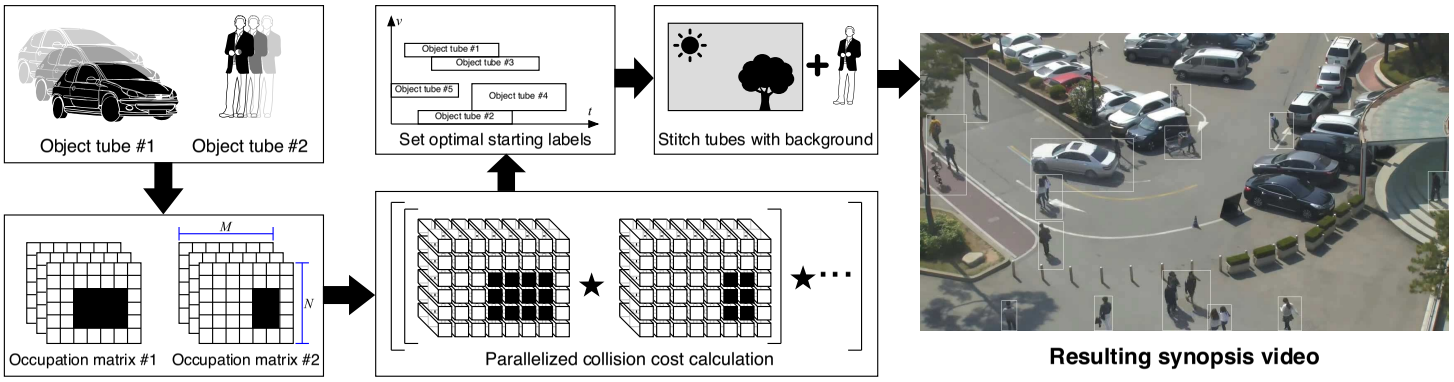


Fig. 3. Flowchart of the proposed online video synopsis framework. At the beginning, foreground of the object tube is reshaped into the 3D occupation matrix. We use this matrix representation to calculate the collision energy fast in conjunction with parallel processing and Fourier transform. Afterwards, we determine optimal starting labels for tubes and stitch tubes with the background to generate a resulting synopsis video. Illustrations of man and vehicle in this figure are created by Lluisa Iborra and Yasser Megahed from the Noun Project.

Binary occupation matrix

The binary occupation matrix $\mathbf{M}^b$ does not allow gray area value to represent the existence of objects; it can only have 1s and 0s. Let assume that the foreground mask of the i^{th} object is denoted as $\mathbf{F}_i(x, y, t) \in \mathbb{B}$; then, $\mathbf{M}_i^b(u, v, t)$ is defined as

$$\mathbf{M}_i^b(u, v, t) = \begin{cases} 1 & \sum_{(x,y)\in C(u,v)} \mathbf{F}_i(x, y, t) \neq 0 \\ 0 & \text{otherwise,} \end{cases} \tag{11}$$

where $C(u, v)$ is a set of 2D coordinates (x, y). Based on (11), to calculate a single element of $\mathbf{M}_i^b$, we need to examine the values of $\mathbf{F}_i$ for every coordinate in $C(u, v)$. The definition of $C(u, v)$ is given by

$$C(u, v) = \left\{(x, y) \mid x \in X(u), y \in Y(v)\right\}, \tag{12}$$

where $X(u)$ and $Y(v)$ are sets of x and y coordinates, respectively.

$$X(u) = \left\{ x \;\middle|\; \left\lfloor \frac{W}{N} u \right\rceil \leq x < \left\lfloor \frac{W}{N} (u + 1) \right\rceil \right\}, \tag{13}$$

$$Y(v) = \left\{ y \;\middle|\; \left\lfloor \frac{M}{H} v \right\rceil \leq y < \left\lfloor \frac{M}{H} (v + 1) \right\rceil \right\}. \tag{14}$$

Due to the condition $\left(\sum_{(x,y)\in C(u,v)} \mathbf{F}_i(x, y, t) \neq 0\right)$ in (11), even a single pixel of $\mathbf{F}_i(x, y, t)$ can produce a response in $\mathbf{M}_i^b(u, v, t)$. Therefore, $\mathbf{M}_i^b$ exaggerates the occupation region of the object tube in the video sequence. An example of the binary occupation matrix gener-

ation is depicted in Fig. 4.

Probabilistic occupation matrix

Since the probabilistic occupation matrix $\mathbf{M}_i^p$ represents the existence of the object tube with continuous values, it can provide more precise information than $\mathbf{M}_i^b$. Each element of $\mathbf{M}_i^p$ is calculated as

$$\mathbf{M}_i^p(u,v,t) = \frac{\sum_{(x,y)\in C(u,v)} \mathbf{F}_i(x,y,t)}{|C(u,v)|}. \tag{15}$$

where $|C(u,v)|$ is a cardinality of $C(u,v)$. In most cases, where $W/N \in \mathbb{N}$ and $H/M \in \mathbb{N}$, $|C(u,v)|$ becomes a constant value.

Binary vs. probabilistic

The main role of the occupation matrix is to provide a spatial approximation of $\mathbf{F}_i$. Since both representations can achieve the objective, selecting the type of the occupation matrix is same as considering the trade-off between the accuracy and computation time. In general, using $\mathbf{M}_i^p$ can produce more compact synopsis video with more computational burden. On the other hand, the scene in the condensed video based on $\mathbf{M}_i^b$ is less complex and can be generated with less computation. Quantitative evaluations regarding the type of the occupation matrix will be given in Section 5.

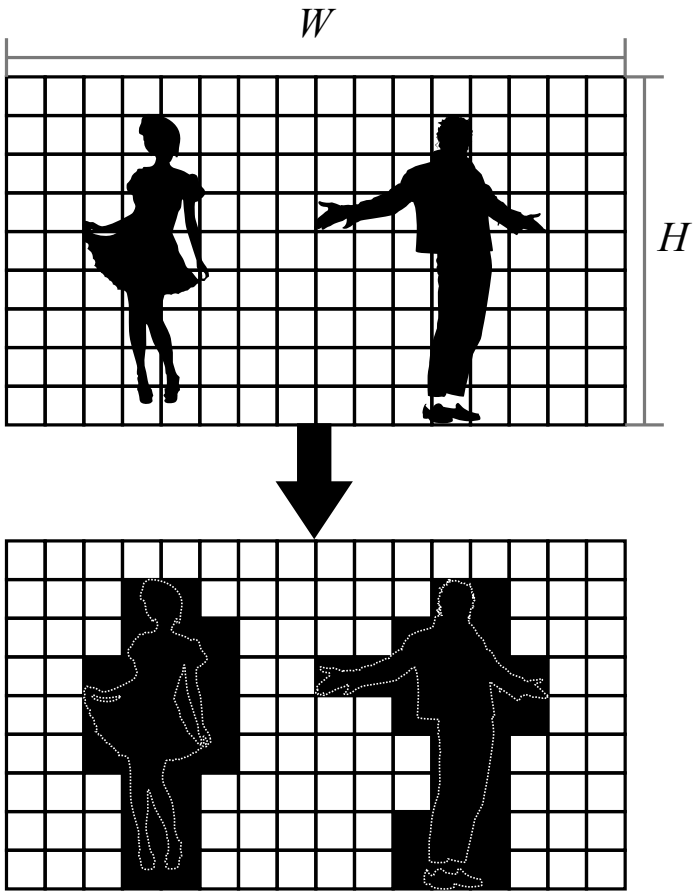


Fig. 4. Example of the binary occupation matrix generation when $W/N = 16$ and $H/M = 9$. The foreground and background of the object are represented in black and white, respectively. Dotted lines in the figure are depicted to show contours of the original object for the readers. Illustrations of the woman and man in this figure are created by Nataliia Lytvyn and Ludovic Gicqueau from the Noun Project, respectively.

3.2 Objective function

For the next step, the collision energy is reformulated with the occupation matrix and a new energy term E_l is introduced to penalize a long condensed video. Then, the final objective function of the proposed tube rearrangement algorithm is defined by considering both E_c and E_l.

Reformulated collision energy

The motivation behind the reformulation of E_c is that the degree of collision between the objects at a certain frame can be calculated as a sum of the element-wise multiplication of two occupation matrices. An example of this computation is depicted in Fig 5 and the redefined $E_c(l_i, l_j)$ is given by

$$E_c(l_i, l_j) = \sum_{u=1}^{M} \sum_{v=1}^{N} \sum_{t=t_{\min}}^{t_{\max}} \mathbf{M}_i(u, v, t) \mathbf{M}_j(u, v, t), \tag{16}$$

where $t_{\min}$ and $t_{\max}$ are minimum and maximum values of the overlapped temporal domain. Detailed calculations of $t_{\min}$ and $t_{\max}$ are

$$t_{\min} = \max(l_i, l_j), \tag{17}$$

$$t_{\max} = \min(T_i + l_i, T_j + l_j), \tag{18}$$

where T_i and T_j are frame lengths of the i^{th} and j^{th} object tubes, respectively.

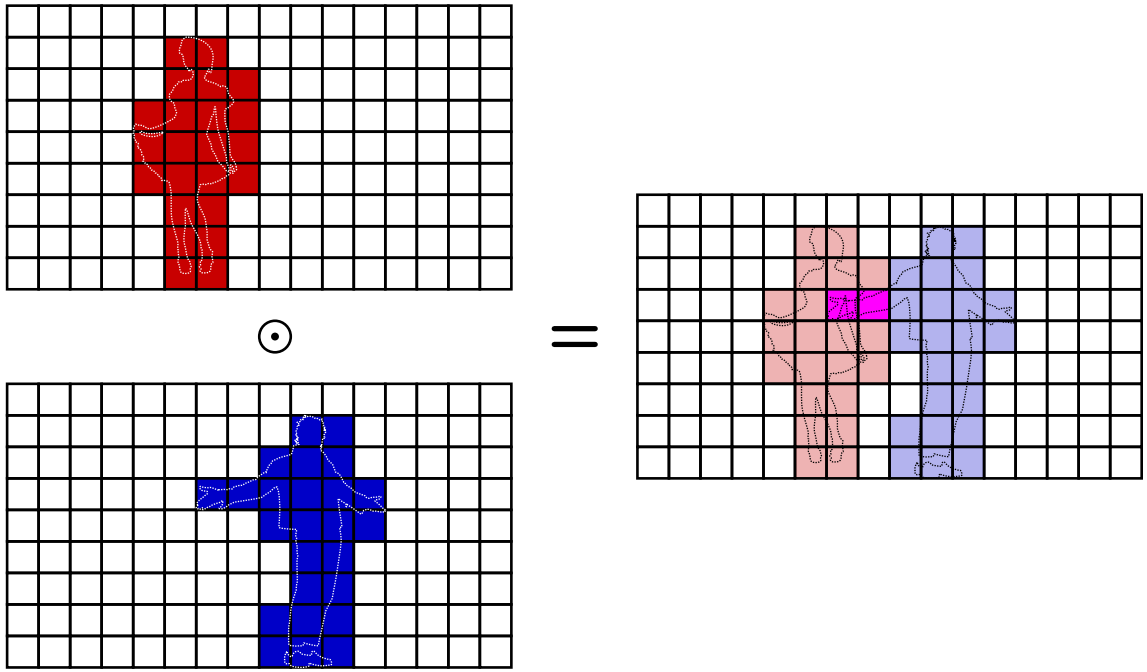


Fig. 5. Example calculation of reformulated collision energy E_c with two binary occupation matrices. Occupied elements in the matrix are colored in red and blue. After the element-wise multiplication, we can see that the objects have two collided elements colored in magenta. In this figure, $\odot$ is an operator for the element-wise multiplication, also known as Hadamard product [].

Length energy

Apart from the existing video synopsis frameworks [] using the fixed length of the synopsis video, the proposed framework adaptively adjusts the length of the condensed video by considering both compactness and complexity. In this regard, the length energy $E_l(l_i, l_j)$ is defined as the frame length of the synopsis video when two object tubes have starting labels of l_i and l_j.

$$E_l(l_i, l_j) = \max(T_i + l_i, T_j + l_j) - \min(l_i, l_j). \tag{19}$$

An objective function $E(l_i, l_j)$ is calculated as a weighted sum of the collision and length energies.

$$E(l_i, l_j) = E_c(l_i, l_j) + \lambda E_L(l_i, l_j), \tag{20}$$

where λ is a weighting parameter adjusting the importance of the length energy. In general, the larger λ generates the shorter but more complex synopsis video; on the other hand, the smaller λ produces the longer but less confused condensed video.

3.3 Optimizing objective function

As in other online video synopsis algorithms [], the proposed tube rearrangement algorithm adopts the stepwise optimization strategy; therefore, starting labels of the object tubes are determined one by one through iterations. At the i^{th} iteration of the optimization, the

starting label of i^{th} object tube l_i is determined as

$$l_i = \arg\min_{l} E(l, L_{i-1}) \text{ subject to } l_i \geq 0, \tag{21}$$

where $L_{i-1} = \{l_1, ..., l_{i-1}\}$ is a set of starting labels determined after $i-1$ iterations. A constraint to the optimization $l_i \geq 0$ is used to alleviate chronological disorder in the synopsis video. In other words, since a negative l_i means that i^{th} tube appear prior to the first tube in the synopsis video, preventing such case increases a chance to keep chronological order of the tubes.

Due to the stepwise optimization strategy, one of two input arguments for E in (21) becomes L_{i-1} instead of a single label as described in (20). In consequence, slight modifications of (16) and (19) are necessary. For the stepwise optimization, the calculation of E_c is modified as

$$E_c(l_i, L_{i-1}) = \sum_{u=1}^{M} \sum_{v=1}^{N} \sum_{t=t^*_{\min}}^{t^*_{\max}} \mathbf{M}_i(u, v, t) \mathbf{M}^*_{i-1}(u, v, t), \tag{22}$$

where $\mathbf{M}^*_{i-1}$ is an accumulated occupation matrix for $i-1$ iterations, and $t^*_{\min}$ and $t^*_{\max}$ are minimum and maximum bounds of shared temporal domain between $\mathbf{M}_i$ and $\mathbf{M}^*_{i-1}$. Moreover, each element of $\mathbf{M}^*_{i-1}$ is defined in the recurrence relation as

$$\mathbf{M}^*_{i-1}(u, v, t) = \mathbf{M}_{i-1}(u, v, t - l_{i-1}) + \mathbf{M}^*_{i-2}(u, v, l^*_{i-2}), \tag{23}$$

where $l^*_{i-2} = \min L_{i-2}$. For the initial condition of (23), $\mathbf{M}^*_1 = \mathbf{M}_1$ and $l^*_1 = l_1 = 0$ are

used. Formal definitions of $t^*_{\min}$ and $t^*_{\max}$ are

$$t^*_{\min} = \max(l_i, l^*_{i-1}) \tag{24}$$

and

$$t^*_{\max} = \min(T_i + l_i, T^*_{i-1} + l^*_{i-1}), \tag{25}$$

where T^*_{i-1} is a frame length of $\mathbf{M}^*_{i-1}$. The length energy for the stepwise optimization is defined as

$$E_l(l_i, L_{i-1}) = \max(T_i + l_i, T^*_{i-1} + l^*_{i-1}) - \min(l_i, l^*_{i-1}). \tag{26}$$

Properties of accumulated occupation matrix

For better understanding of the stepwise optimization process, we will discuss about properties of the accumulated occupation matrix $\mathbf{M}^*$. According to the type, the occupation matrix $\mathbf{M}$ can have either Boolean or continuous values in the range from 0 to 1. On the other hand, $\mathbf{M}^*$ is computed by adding two matrices as described in (23); therefore, each element of $\mathbf{M}^*$ belongs to either $\mathbb{N}_0$ or $\mathbb{R}_{\geq 0} = \{x \in \mathbb{R} \mid x \geq 0\}$. By utilizing $\mathbf{M}^*$, we can represent occupation and collision states of more than two objects on the single matrix.

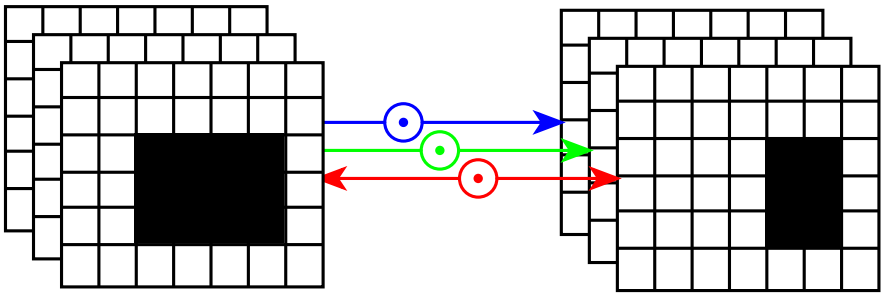
3.4 Parallelized optimization

Even though $\mathbf{M}$ provides an efficient way of representing the object tubes and E_c can be computed easily with the element-wise multiplication, optimization of E_c can further be accelerated by using both parallel processing and cross-correlation of two occupation matrices in the temporal domain. Prior to define the cross-correlation, let assume that two occupation matrices overlap by at least one frame in the temporal domain. Without this restriction, E needs to be evaluated for every possible l value. Then, the parallelized version of E_c in (22) is defined as

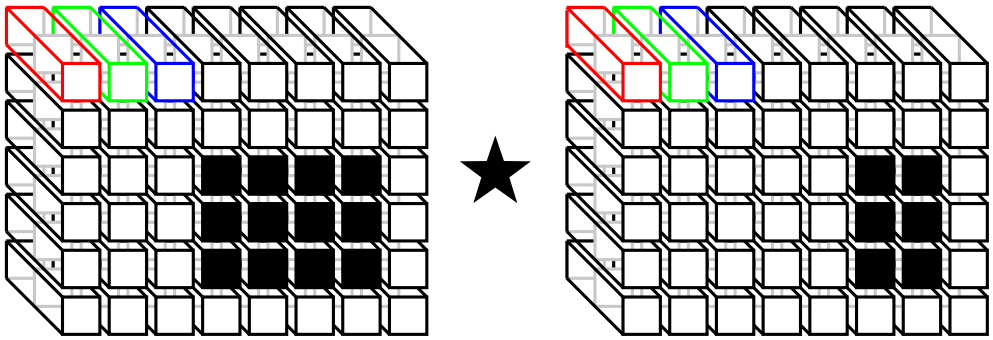
$$\begin{aligned} E_c(l_i, L_{i-1}) &= \sum_{u=1}^{M}\sum_{v=1}^{N} \mathbf{M}_i \star \mathbf{M}_{i-1}^{*}(u, v, l_i - l_{i-1}^{*}) \\ &= \sum_{u=1}^{M}\sum_{v=1}^{N}\sum_{t=-\infty}^{\infty} \mathbf{M}_i \mathbf{M}_{i-1}^{*}(u, v, t + l_i - l_{i-1}^{*}), \end{aligned} \tag{27}$$

where $\star$ is an operator for the cross-correlation.

The motivation behind the conversion from (22) to (27) is illustrated in Fig. 6. From (22), if we take the spatial coordinate into the consideration first, the 3D element-wise multiplication can be thought as a series of 2D Hadamard products in temporal domain as shown in Fig 6a. On the other hand, if we consider the temporal domain first, the operation becomes $\mathcal{M} \times \mathcal{N}$ 1D cross-correlations as illustrated in Fig 6b. This difference may seem to be minor but it is important when we consider some tricks to accelerate the operation. The computational burden of multiple 1D cross correlations can be reduced by Fast Fourier Transform (FFT) [16] in conjunction with parallel processing. A detailed procedure of the proposed tube rearrangement algorithm is presented in Algorithm 1.



(a) 2D Hadamard products



(b) 1D cross-correlations

Fig. 6. Two ways of calculating E_c. All of occupation matrices in this figure have $6 \times 8 \times 3$ spatio-temporal resolution. E_c can be calculated by using (a) Hadamard products between two sets of frames, and (b) 1D cross correlations between 48 pairs of 1D signals. Three primitive colors (red, green, and blue) in this figure is used to show example correspondences.

Algorithm 1 Proposed tube rearrangement algorithm

Input: $\mathbf{M}_i, i = 1, ..., N$

Output: $L_N = \{l_1, ..., l_N\}$

$\mathbf{M}_1^* = \mathbf{M}_1$, $l_1^* = l_1 = 0$, $L_1 = \{l_1\}$

for $i = 2$ to N **do**

Calculate $\mathbf{M}_i \star \mathbf{M}_{i-1}^*$ using FFT and parallel processing

Find a local optimum starting label l_i by using (21)

Calculate $\mathbf{M}_i^*$ from $\mathbf{M}_i$ and $\mathbf{M}_{i-1}^*$ by using (23)

$L_i = L_{i-1} \cup l_i$

$l_i^* = \min L_i$

end for

return L_N

4 Online video synopsis framework

The proposed tube rearrangement algorithm is based on the online framework. Similar to existing online frameworks [], the proposed framework consists of four stages: background modeling, object tube generation, tube rearrangement, and object stitching. Among them, three components, except for the tube rearrangement, will be explained in detail.

4.1 Background modeling

There are numerous choices for modeling the background, since this field of research has been studied for decades. According to Bouwmans *et al.* [], deep learning based background modelings can be categorized into three groups: deep encoder-decoder, convolutional neural networks, and generative adversarial network (GAN).

Even though deep learning based approaches show better foreground segmentation results than conventional statistical background modeling approaches [], their computational burden are hard to be ignored. Therefore, in this dissertation, the proposed framework utilizes a well-known Gaussian mixture model [25] to separate the foreground of the objects from the background. In addition, this modeling process can be accelerated by using the GPGPU.

4.2 Object tube generation

After the background modeling stage, we can get

4.3 Object stitching

In the object tube generation stage, foregrounds that belong to the same object are associated by using the Hungarian algorithm [26] to generate the object tubes. Then, the generated object tubes are stored and maintained in a queue. When the size of the queue exceeds K, the starting labels of the object tubes in the queue are determined by the tube rearrangement algorithm, and the entire contents of the queue are cleared. After determining the starting

	Video Clips	Resolution	# Frame	# Tube
1	Parking lot square I	1280×720	44,057	650
5	Parking lot square II	640×360	107,946	271
2	Crossroad I	640×360	85,766	291
3	Crossroad II	640×360	106,459	937
4	Library lobby	1280×720	49,679	316
6	Subway station plaza	640×360	107,876	1038

Table 1. List of test sequences used in the experiments

labels, the corresponding object tubes are stitched back into the background to generate a small portion of the synopsis video. For a stitching algorithm, the Poisson image editing [27] is utilized.

Because the aforementioned process generates the synopsis video in a stepwise fashion, a discontinuity of motion flow problem [15] occurs when merging small portions of the synopsis video into the complete one. This problem can be solved by considering the tails of the object tubes in the previous step during the current rearrangement step.

5 Experimental Results

In this section, the performance of the proposed tube rearrangement algorithm is evaluated by using four metrics: frame condensation ratio (FR), compact ratio (CR), overlap ratio (OR), and running time (RT). The experiments are conducted on a 4-core 4.0 GHz computer with 32 GB of memory. For the test sequences, six video clips are captured at four different

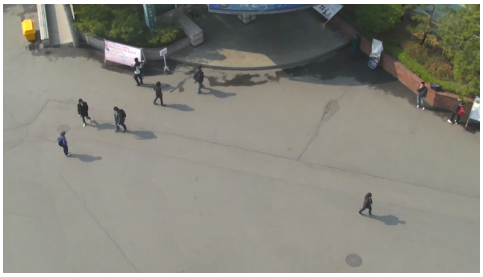
(a) Parking lot square I

(b) Parking lot square II

(c) Crossroad I

(d) Crossroad II

(e) Library lobby



(f) Subway station plaza

Fig. 7. Examples of the test sequences. All sequences were captured at Hanyang university, Seoul, Korea.

places: a parking lot square, a crossroad, a library lobby, and a subway station plaza. Detail characteristics of the test sequences are summarized in Table 1. Some examples of the test sequences are depicted in Fig. 7. The experiments are designed to 1) prove the efficiency of the parallelized tube rearrangement and 2) compare performances of several different online tube rearrangement algorithms.

5.1 Efficiency of parallelized tube rearrangement

To show the efficiency of our proposed parallelization scheme, parallelized and non-parallelized versions of the proposed algorithm are compared in terms of running time. A result of the comparison is illustrated in Fig. 8. According to the results, the parallelized version is approximately 2.5 times faster than the non-parallelized version. Because the current parallelized version is based on CPU multi-threading, the proposed algorithm can be further efficiently computed with GPU-based parallelization.

5.2 Comparisons of different online tube rearrangement algorithms

We have reproduced three recently introduced online tube rearrangement algorithms to compare and analyze the performance of the proposed algorithm. For fair comparisons, the required parameters of existing algorithms are set the same as in the original papers. Note that existing algorithms have not employed any spatial subsampling. Results of the

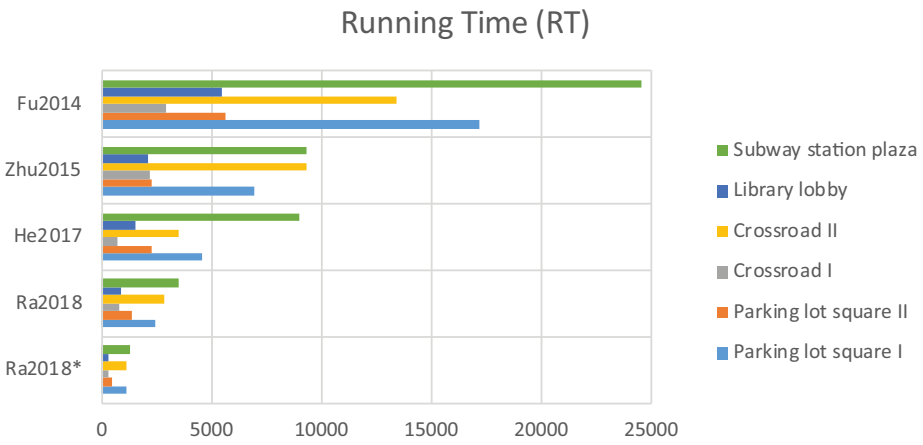


Fig. 8. Running times of the different tube rearrangement algorithms for six test sequences. An asterisk (*) indicates the parallelized version of the proposed algorithm.

comparisons are illustrated in figs. 8 to 11. For FR, the proposed algorithm achieved the second-best performance in overall. For CR, the algorithm of Zhu *et al.* [13] performed the best, while the others do not show significant differences. On the other hand, regarding RT and OR, the proposed algorithm outperformed other algorithms by a wide margin. Especially for RT, even a non-parallelized version has an advantage over existing algorithms. The experiments show that the proposed algorithm can efficiently handle occlusions far better than existing ones, even with a large number of objects.

6 Conclusion

This letter proposes the parallelized tube rearrangement algorithm for online video synopsis. For reducing the computational bottleneck caused by pairwise energy terms, the proposed

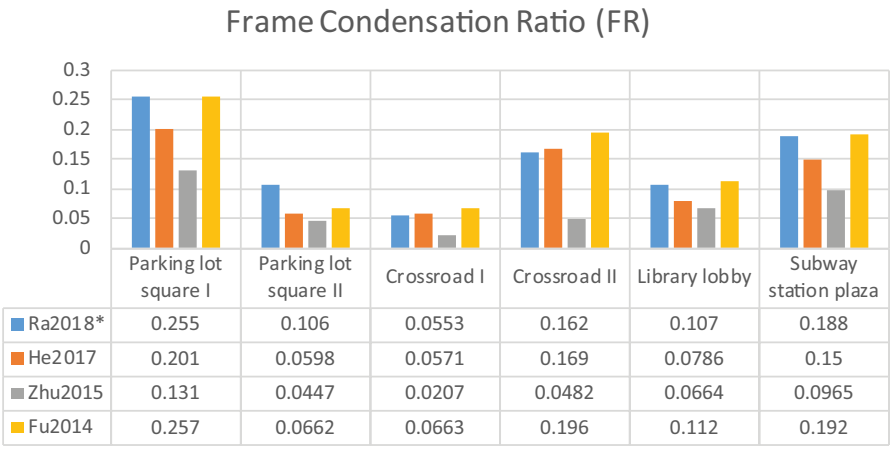


Fig. 9. Frame condensation ratios of different tube rearrangement algorithms for six test sequences.

algorithm concurrently computes the collision energy defined as multiplications of 3D occupation matrices by using an FFT. Throughout the experiments, the proposed algorithm outperformed existing algorithms in terms of computation time and overlap ratio, while its other performance metrics were comparable with those of other algorithms. Our future work will involve running the proposed algorithm on a GPU.

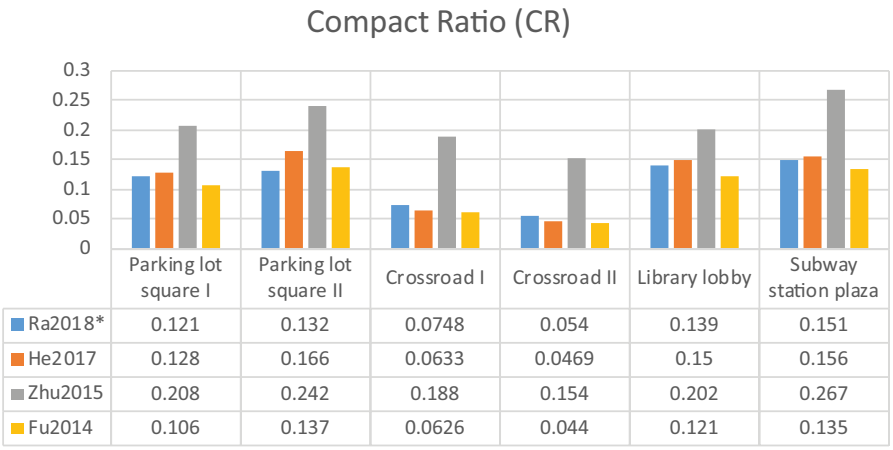


Fig. 10. Compact ratios of the different tube rearrangement algorithms for six test sequences.

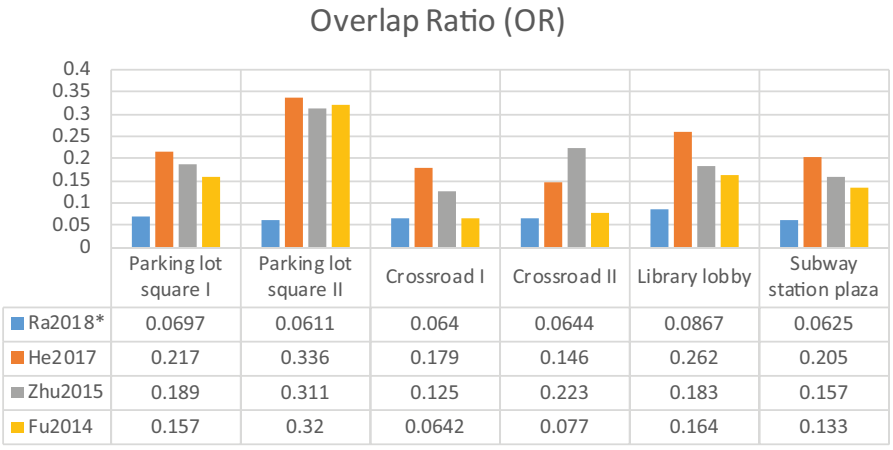


Fig. 11. Overlap ratios of the different tube rearrangement algorithms for six test sequences.

BIBLIOGRAPHY

[1] Y. Pritch, S. Ratovitch, A. Hendel, and S. Peleg, "Clustered Synopsis of Surveillance Video," in *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*. IEEE, sep 2009, pp. 195–200. [Online]. Available: http://ieeexplore.ieee.org/document/5280098/

[2] M. Smith and T. Kanade, "Video skimming and characterization through the combination of image and language understanding techniques," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Soc, 1998, pp. 775–781. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=609414 http://ieeexplore.ieee.org/document/609414/

[3] N. Petrovic, N. Jojic, and T. S. Huang, "Adaptive Video Fast Forward," *Multimedia Tools and Applications*, vol. 26, no. 3, pp. 327–344, aug 2005. [Online]. Available:

http://link.springer.com/10.1007/s11042-005-0895-9

[4] B. Höferlin, M. Höferlin, D. Weiskopf, and G. Heidemann, "Information-based adaptive fast-forward for visual surveillance," *Multimedia Tools and Applications*, vol. 55, no. 1, pp. 127–150, oct 2011. [Online]. Available: http://link.springer.com/10.1007/s11042-010-0606-z

[5] A. Rav-Acha, Y. Pritch, and S. Peleg, "Making a Long Video Short: Dynamic Video Synopsis," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06)*, vol. 1. IEEE, 2006, pp. 435–441. [Online]. Available: http://ieeexplore.ieee.org/document/1640790/

[6] Y. Pritch, A. Rav-Acha, A. Gutman, and S. Peleg, "Webcam Synopsis Peeking Around the World.pdf," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.

[7] Y. Pritch, A. Rav-Acha, and S. Peleg, "Nonchronological Video Synopsis and Indexing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1971–1984, nov 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4444355/

[8] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, feb 2004. [Online]. Available: http://ieeexplore.ieee.org/document/1262177/

[9] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671–680, may 1983. [Online]. Available: http://www.sciencemag.org/cgi/doi/10.1126/science.220.4598.671

[10] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*, 3rd ed. MIT press, 2009.

[11] C.-R. Huang, P.-C. J. Chung, D.-K. Yang, H.-C. Chen, and G.-J. Huang, "Maximum <italic>a Posteriori</italic> Probability Estimation for Online Surveillance Video Synopsis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 8, pp. 1417–1429, aug 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6748870/

[12] Shikun Feng, Zhen Lei, Dong Yi, and S. Z. Li, "Online content-aware video condensation," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2012, pp. 2082–2087. [Online]. Available: http://ieeexplore.ieee.org/document/6247913/

[13] Jianqing Zhu, Shikun Feng, Dong Yi, Shengcai Liao, Zhen Lei, and S. Z. Li, "High-Performance Video Condensation System," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 7, pp. 1113–1124, jul 2015. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6928452 http://ieeexplore.ieee.org/document/6928452/

[14] Y. He, Z. Qu, C. Gao, and N. Sang, "Fast Online Video Synopsis Based on Potential Collision Graph," *IEEE Signal Processing Letters*, vol. 24, no. 1, pp. 22–26, jan 2017.

[Online]. Available: http://ieeexplore.ieee.org/document/7762044/

[15] W. Fu, J. Wang, L. Gui, H. Lu, and S. Ma, "Online video synopsis of structured motion," *Neurocomputing*, vol. 135, pp. 155–162, jul 2014. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0925231214000666

[16] A. V. Oppenheim and R. W. Schafer, *Discrete Time Signal Processing*, 3rd ed. Pearson, 2010.

[17] Y. Nie, H. Sun, P. Li, C. Xiao, and K.-L. Ma, "Object Movements Synopsis via Part Assembling and Stitching," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 9, pp. 1303–1315, sep 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6702519/

[18] X. Zhu, J. Liu, J. Wang, and H. Lu, "Key observation selection-based effective video synopsis for camera network," *Machine Vision and Applications*, vol. 25, no. 1, pp. 145–157, jan 2014. [Online]. Available: http://link.springer.com/10.1007/s00138-013-0519-8

[19] A. Mahapatra, P. K. Sa, B. Majhi, and S. Padhy, "MVS: A multi-view video synopsis framework," *Signal Processing: Image Communication*, vol. 42, pp. 31–44, mar 2016. [Online]. Available: http://dx.doi.org/10.1016/j.image.2016.01.002 http://linkinghub.elsevier.com/retrieve/pii/S0923596516000059

[20] S.-z. Wang, Z.-y. Wang, and R.-m. Hu, "Surveillance video synopsis in the compressed domain for fast video browsing," *Journal of Visual Communication and*

Image Representation, vol. 24, no. 8, pp. 1431–1442, nov 2013. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1047320313001818

[21] Rui Zhong, Ruimin Hu, Zhongyuan Wang, and Shizheng Wang, "Fast Synopsis for Moving Objects Using Compressed Video," *IEEE Signal Processing Letters*, vol. 21, no. 7, pp. 834–838, jul 2014. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6805196 http://ieeexplore.ieee.org/document/6805196/

[22] X. Li, Z. Wang, and X. Lu, "Surveillance Video Synopsis via Scaling Down Objects," *IEEE Transactions on Image Processing*, vol. 25, no. 2, pp. 740–755, feb 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7353185/

[23] Zhuang Li, P. Ishwar, and J. Konrad, "Video Condensation by Ribbon Carving," *IEEE Transactions on Image Processing*, vol. 18, no. 11, pp. 2572–2583, nov 2009. [Online]. Available: http://ieeexplore.ieee.org/document/5156267/

[24] K. Li, B. Yan, W. Wang, and H. Gharavi, "An Effective Video Synopsis Approach with Seam Carving," *IEEE Signal Processing Letters*, vol. 23, no. 1, pp. 11–14, jan 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7312927/

[25] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 2, no. 2. IEEE, 2004, pp. 28–31 Vol.2. [Online]. Available: http://ieeexplore.ieee.org/document/1333992/

[26] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, mar 1955. [Online]. Available: http://doi.wiley.com/10.1002/nav.3800020109

[27] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Transactions on Graphics*, vol. 22, no. 3, p. 313, 2003. [Online]. Available: http://portal.acm.org/citation.cfm?doid=882262.882269