Foreground-Background Segmentation Based Surveillance Video Coding

Jiyong Yu*, Luheng Jia*§, Yifan Zang*, Zhaoyang Yu*, Shuyuan Zhu†, Li Song‡ and Kebin Jia*
* Beijing University of Technology, Beijing, China
† University of Electronic Science and Technology of China, Sichuan, China
‡ Shanghai Jiao Tong University, Shanghai, China
§ Corresponding E-mail: luhengjia@bjut.edu.cn

***Abstract*—In order to cope with the increasing data volume of surveillance video, we propose a surveillance video coding method based on foreground-background segmentation. The proposed method extends the conventional encoding framework by incorporating four key components including the auto-SAM2, the frame segmentation module, the complete background constructor, and the frame compositor. The auto-SAM2 first achieves an automatic pixel-accurate mask of the foreground (FG) and background (BG) regions based on the traditional segmentation anything 2 (SAM2) that avoids manual input of prompts. The frame segmentation module converts pixel level mask into block level mask. The blocks in original frames are categorized into dual-sequence pipelines including a FG sequence and a BG frame sequence. The complete BG constructor iteratively complements and renews a complete BG frame by comparing the BG content of current frame blocks with the established complete background (CBG) of previous frame blocks. Meanwhile, the FG sequence and the CBG sequence are sent to two separate encoders, respectively. After decoding, an FG frame is composited with a CBG frame to reconstruct the final video frame. The proposed method achieves improved coding efficiency of approximately -14.2% BD-rate on average compared to HEVC encoder.**

I. Introduction

The widespread adoption of intelligent surveillance systems has led to an exponential increase in video data transmission and storage volume [1]. A single high-definition camera can generate tens of gigabytes of raw video data per day. Efficiently processing various types of surveillance video using tailored approaches has become a key challenge in video compression research. Although traditional methods for general video compression (e.g., H.264/AVC [2] , H.265/HEVC [3]), significantly reduce the bitrate in video transmission or storage, these encoders fail to fully exploit the characteristics of surveillance video that relatively static background (BG) which often receives less attention, while the foreground (FG) that attracts greater focus require higher quality.

Many researchers have proposed encoding strategies for surveillance videos. The methods proposed in [4], [5] adopt the traditional approach of constructing a BG frame as a reference, which struggles to efficiently handle the coding units (CU) containing in both FG and BG content. When these CUs use block located in BG region of reference frame for inter prediction, the FG part of the CU lacks accurate reference, resulting in a substantial increase in bitrate. In contrast, when using blocks in FG of reference frame for inter prediction, the BG region of current block cannot be accurately predicted. The work in [6] leverages the average of the first Group of Pictures (GOP) images as a BG hyperprior. For subsequent GOPs, the BG hyperprior frame is subtracted from the current GOP, and the resulting difference is input into the encoder. However, this method does not update the BG hyperprior over time, causing the difference frame containing a substantial noise and residual BG content. Meanwhile, if there is a change in lighting or FG, the BG hyperprior does not respond timely leading to reduced coding efficiency.

In general, there are two major drawbacks in previous surveillance video compression methods. The first is the high bitrate caused by inaccurate prediction of the region across BG and FG, which is intrinsically due to the lack of efficiency of block based single-directional motion estimation dealing with occlusion problems under surveillance video coding scenario. The second drawback is the lack of accuracy of the generated BG frame due to belated updating strategy that fails to providing efficient prediction. We propose a novel surveillance video coding scheme that handles the drawbacks by conducting decorrelation in the sequence level, specifically by separating and independently compress the FG and BG. For the FG sequence, block based inter prediction can be accurately performed leading to small residual and hence better reconstucted quality under given bits budget. Additionally, the BG is constructed and updated frame by frame to gradually form a complete BG (CBG) sequence that is close to the true BG. The updated CBG is progressively providing increasing referencing performance. While the final CBG will avoid the inaccurate prediction caused by occlusion that leads to small residual for the rest of the BG frames. On the decoder side, the reconstructed FG frame is overlaid onto the reconstructed CBG frame to construct the full frame. Two key techniques are hence important for the proposed scheme including the FG segmentation method and the CBG construction strategy.

Firstly, to achieve accurate inter prediction for the FG sequence and CBG frame close the true BG, precise segmentation of foreground-background is required. Traditional segmentation methods model the BG and generate the corresponding FG mask. For example, the method proposed in [7] employs a Gaussian mixture model (GMM) to detect FG and morphological operations and filtering to process the FG mask. In addition

This work was supported by National Natural Science Foundation of China under Grant 61901012 and Science and Technology Commission of Shanghai Municipality under Grant 22DZ2229005. (Corresponding author: Luheng Jia.)

979-8-3315-7206-8/25/$31.00

to GMM, the K-nearest neighbor classification (KNN) [8] is also widely used for FG object detection. For example, KNN combined with a Gaussian filter employed in [9] can address the issue of blurred object contours in traditional KNN. Traditional FG recognition algorithms can quickly and effectively identify moving objects and are also capable of utilizing the event correlation of videos. However, these methods do not effectively prevent large objects and slow-moving objects from intruding into the BG region, nor can they avoid BG regions intruding into the FG region due to factors such as lighting changes or excessive noise. It is difficult to ensure the accuracy and segmentation of FG-BG and the cleanliness of the BG and FG sequence.

Thanks to the development of neural networks and deep learning, a large number of neural network models have been applied to image recognition and segmentation tasks [10]. These models are capable of accurately detecting both object categories and edges, and this technical approach is termed instance segmentation [11]. U-Net [12] and its extensions (for example, Attention U-Net [13] and U-Net plus [14]) have been widely applied in object detection and recognition tasks within the field of computer vision, due to their unique encoder-decoder architecture and skip connection design. Mask R-CNN, which is proposed in [15], builds on the foundational principles of Faster R-CNN [16] while advancing its capabilities through two key innovations: it employs the ResNet-FPN (feature pyramid network) architecture for feature extraction and introduces an additional mask prediction branch to enable instance segmentation. Although these methods effectively address the shortcomings of traditional FG segmentation, their video processing strategies still treat each frame as an independent static image without leveraging temporal correlation. Consequently, this leads to discontinuous FG segmentation results and a significant increase in the number of intra-blocks processed by the encoder.

Segment Anything 2 (SAM2) [17] integrates the strengths of traditional FG segmentation and deep-learning-based instance segmentation: temporal correlation and spatial correlation. SAM2 is the first unified model capable of real-time, promptable image and video object segmentation. However, SAM2 requires manual input of prompts to mark target instances and cannot be used directly for surveillance video encoding. Therefore, the proposed method integrates an automatic prompt detection mechanism into SAM2, leading to the segmentation scheme as auto-SAM2 that is more suitable for video coding scenario.

Secondly, CBG construction is important for video compression. The GMM proposed in [7] can simulate BG through a multimodal distribution. D.Comanici and P.Meer in [18] proposes a non-parametric BG construction method utilizing the mean shift technique. Antoine Manzanera and Julien Richefeu in [19] estimates two orders of temporal statistics per pixel and uses a Markov model to establish a multiple observation field. The CBG frames constructed by the above methods are synthesized rather than real, and hence providing inefficient referencing performance. Specifically, for the previously occluded BG regions that are reappearing in the current frame of static BG sequences with still camera shooting, the synthesized BGs using previous methods are blurred or contain perceptual artifacts. To solve the problem, our proposed method fully leverages the pixel-level BG mask output by the frame segmentation module and constructs the complete real BG through the CBG constructor in block-level. The close-to-real BG updated in real-time not only efficiently deal with the occlusion problem, but also provides higher prediction accuracy than the long-term BG frame.

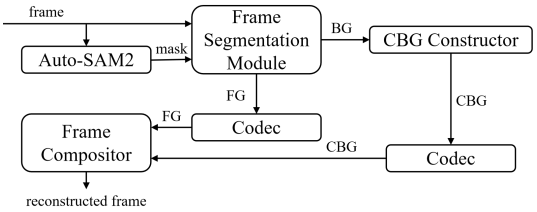


Fig. 1: The overview structure of the proposed method.

The remainder of this paper is structured as follows. Section II presents the proposed method. Section III presents the experimental results and analysis. Section IV is a conclusion.

II. The Proposed Method

A. Overview

The general framework of the proposed method is shown in "Fig. 1". With the original frame sequence as input, the proposed auto-SAM2 automatically identifies FG instances and generates the corresponding masks for FG instances. After the pixel-level masks are obtained, the frame segmentation module divides the original sequence into FG sequence and BG sequence at block level to be compatible with video encoder. The BG sequence supports generating the CBG sequence gradually using the CBG constructor with carefully designed updating strategy. Then, two separate encoders compress the FG sequence and the CBG sequence, respectively. Finally, on the decoder side, FG and CBG are combined to form the reconstructed frame sequence exploiting the frame compositor module.

B. Auto-SAM2

The proposed auto-SAM2 comprises three components including a prompt maker, a GOF splitter and a SAM2 module. The prompt maker automatically outputs the FG boxes and uses the central points of the boxes as the prompts. With the obtained prompts, the GOF splitter takes frames with changes in the number of prompts in a group of video frames (GOF) as key frame and other frames as normal frames. According to the classification, a GOF is hence divided into a group of key frames (GOK) and a group of normal frames (GON). Then the splitter

(a) the points prompt: red for negative and green for positive

(b) the output mask

Fig. 2: The effect and output of points prompt applied on CUHK sequence.

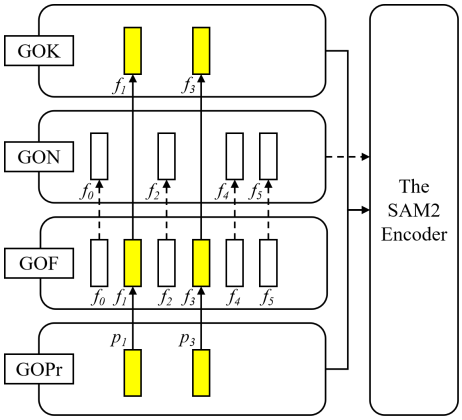


Fig. 3: The GOF Splitter.

sends the GOK and corresponding prompts to the SAM2 module, which extracts features from prompts and key frames to form masks and saves them for future reference.

The SAM2 requires a GOF and a group of prompts (GOPr) as input, where the prompt serves to provide guidance for the target instance, which has four candidate forms including positive/ negative points (”Fig. 2”), box, mask, and skipped. However, prompts are annotated manually in the original SAM2 [17]. In order to enable SMA2 to automatically recognize the FG instances, in the proposed auto-SAM2 scheme, the prompt maker module is leveraged to automatically generate prompts, which uses an ultra-lightweight recognition model YOLACT in [20] to identify FG instances and generate bounding boxes of FG instances. To more accurately locate the objects, the prompt maker uses the center point of each bounding box as the initial point. Then it calculates the number of points in each frame. If the number of points in frame *i* exceeds that in frame *i*-1, frame *i* is designated as a key frame, which means new FG instances that need to be recognized appear in frame *i*. Only points in the key frames are set as prompts and used in SAM2 module .

To identify FG instances more accurately, auto-SAM2 maximizes the temporal correlation of each frame in the surveillance video. The GOF splitter in ”Fig. 3” divides each GOF into GOK (group of key frames) and GON (group of normal frames). The GOK and GOPr are first fed to the SAM2 module for encoding. Afterward, the GON are sent to the SAM2 module as well. With the specifically designed frames processing order, forward prediction of FG instance masks can be conducted using the prompt of the current frame to predict the corresponding mask in its previous frames.

The SAM2 module shown in ”Fig. 4” extracts the GOK and GOPr features from the input and generates a mask indicating the corresponding instance by labeled prompt. These features will be stored in the memory bank for use by other frames in this GON. The memory management system shown in ”Fig. 5” is the most important part that distinguishes the proposed auto-SAM2 scheme from other instance segmentation networks. The memory encoder downsamples mask and undergoes element-wise addition with frame embedding, followed by fusion through a lightweight convolutional layer, ultimately completing the feature fusion process. The memory bank is implemented through a a first-in-first-out caching mechanism storing recent frame

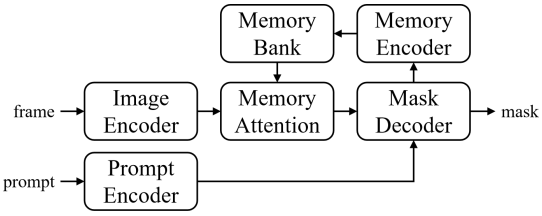


Fig. 4: The structure of the SAM2 module.

memories, comprising two components including the historical bank stores information of the GOK features and the information of *N* coded normal frames, while the prompt bank holds up to *M* frames of prompt information. To adaptive to video coding task, this mechanism is optimized for video object segmentation, continuously retaining the initial frame memory and recently unprompted frames to support continuous video processing requirements. Memory attention aims to achieve spatiotemporal context modeling by integrating current frame features, historical frame prediction results, and prompts. Its architecture employs stacked transformer blocks. Each layer sequentially performs self-attention and cross-attention, and utilizes vanilla attention operations [21] for efficient optimization. Finally, the SAM2 module generates a group of masks that are leveraged by the frame segmentation module.

C. Frame Segmentation Module

Although the proposed auto-SAM2 can already fully leverage both the temporal correlation of videos and instance features, the generated pixel level mask still suffers from unsatisfactory precision. If this pixel level mask is directly used for subsequent BG construction, a significant portion of the FG at the edge of the instance would be erroneously incorporated and updated within the BG. These unrecognized FG edges consist of pixels randomly located in the FG and BG, which causes abrupt change in the same object of neighboring frames, leading to inaccurate inter prediction and high bits consumption after encoding. To solve this problem, the frame segmentation module trades off between the recall and precision [22] of the BG by segmenting the frame into FG and BG in block level as follows.

$$block = \begin{cases} B^{FG}, & \text{if } \left(n_t/(bs)^2 \geq 5\%\right) \\ B^{BG}, & \text{else} \end{cases} \quad (1)$$

where *bs* is the side length of a square block (e.g. 8, 16, 32 and 64) to represent the size of a block which is determined according to the video encoder setting. and n_t is the number of true labels in the same block of pixel-level mask. If the proportion of real labels in one block is higher than 5%, this *block* is classified as a FG block B^{FG} . Otherwise, the *block* is a BG block B^{BG}. The classified B^{FG} and B^{BG} are combined to form FG frame and BG frame, respectively.

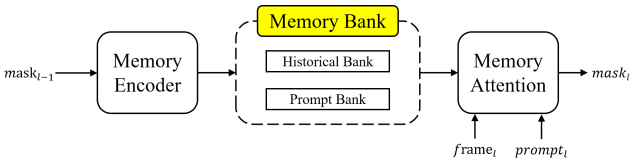
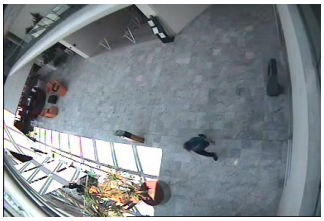


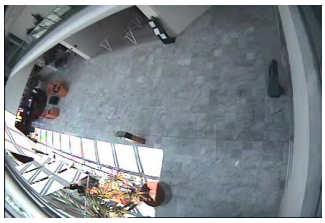
Fig. 5: The memory management system of the auto-SAM2.



(a) The original frame

(b) the FG frame

(c) The BG frame



(d) The CBG frame

Fig. 6: The effect of the frame segmentation module and the CBG constructor in CAVIAR_Browse2:

D. Complete Background Constructor

The main aim of the complete background constructor is to form a clean and close-to-real CBG frame progressively using the BG blocks in sequential frames as shown in "Fig. 6" , which is constructed following the strategy as

$$B_{i,j,l}^{CBG} = \begin{cases} B_{i,j,l}^{BG}, & \text{if } (MAD > thres) \\ B_{i,j,l-1}^{CBG}, & \text{else} \end{cases} \quad (2)$$

where $B_{i,j,l}^{BG}$ and $B_{i,j,l}^{CBG}$ are the BG block and CBG block at position (i, j) in l-th frame, respectively. And *MAD* is the mean absolute difference [23] in one block estimating difference between the current BG block$B_{i,j,l}^{BG}$ and collocated CBG block in previous CBG frame $B_{i,j,l-1}^{CBG}$ calculated as

$$MAD = \frac{\sum_{x=i\times bs}^{(i+1)\times bs} \sum_{y=j\times bs}^{(j+1)\times bs} \left| B_{i,j,l}^{BG}(x,y) - B_{i,j,l-1}^{CBG}(x,y) \right|}{bs^2} \quad (3)$$

where $B_{i,j,l}^{BG}(x,y)$ and $B_{i,j,l-1}^{CBG}(x,y)$ is the pixel value of BG and CBG at position (x, y) in the block.

Additionally, the *thres* is the threshold that determines whether to update $B_{i,j,l}^{CBG}$ jointly considering segmentation block size and encoder parameter as follows.

$$thres = (base\_thres + (1 + step \times QP)) \times (bs/16)^2 \quad (4)$$

in which *base_thres* is a basic threshold when *bs* is 16, *QP* is the quantization parameter of the encoder and *step* is an empirical value used to scale the *QP* in calculating threshold value. The *thres* is adjusted based on *bs* and *QP* to adapt to applications with different target bitrates and instances sizes in the surveillance video.

It is interesting to notice that the CBG constructing process eliminates large and unimportant temporal changes, which is similar to low-pass filtering. Therefore, BG sequence compression is efficient with reduced bitrate and uncomprimised quality.

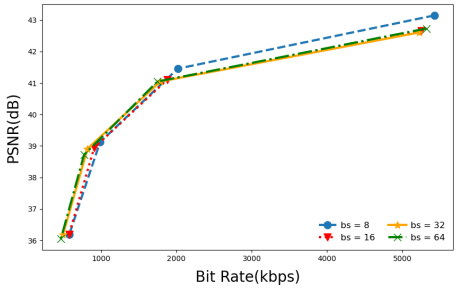


Fig. 7: the result of Class_E_KristenAndSara encoded with QPs 17, 22, 27, and 32 using different *bs* by HEVC in the proposed method.

E. Codec

The codec can be arbitrarily selected from various video compression standards, such as AVC [2] and HEVC [3], etc. The block size measurement bs will vary depending on the coding mode, the coding unit (CU) dimension employed by the encoder and the content of the video. For example, in Fig 7 using HEVC for encoding, the optimal block size in Class_E is 8×8 for higher encoding quality when QP=17 and QP=22, while the 64×64 sized block is used for lower quality encoding with QP=27 and QP=32.

F. Frame Compositor

The frame compositor is to combine FG and CBG at the decoding end. Since all block in the current frame will be divided into FG or CBG, the compositor directly covers the FG on CBG to generate the reconstructed frame.

III. Experimental Results

A. Experiment Setting

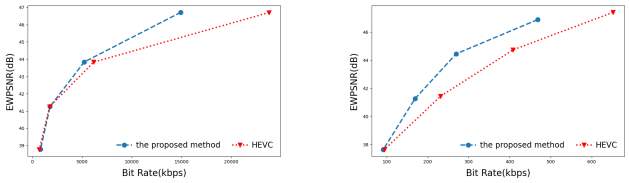
The experiment utilizes video clips from the widely used surveillance video datasets EWAP, CUHK_Avenue and CAVIAR [24] for testing. The HEVC Clase_E sequence is also employed as a indoor surveillance video with large objects. The proposed method uses x265 encoder [25] with placebo encoding mode and fixed QPs as 22, 27, 32 and 37. Referencing structure IPPP is used. The parameter *base_thre* is set to 350, and *step* is set to 0.333. The comparison method is to use only the x265 encoder, and its settings are the same as the proposed method.

B. Measurements

For surveillance videos, human attention tends to be directed more toward the FG than toward the BG. To better capture the quality characteristics of the surveillance footage, we select the EyeTracking Weighted PSNR (EWPSNR) [26] and the Bjøntegaard Delta Bit Rate (BD-rate) [27] as metrics to evaluate the subjective quality of videos.

C. Results and Analysis

The results of some clips are shown in "Fig. 8" and in Table I. The proposed scheme achieves better rate-distortion performance compared to x265 encoder with average BD-rate of approximately -14.2%. The proposed scheme achieves significant coding gain on CAVIAR dataset and EWAP_hotel consist of



(a) Class_E_KristenAndSara (b) EWAP_hotel

Fig. 8: The R-D curves using QP of 17, 22, 27, 32

overhead-view, high-noise video clips. It is due to that the proposed CBG updating has denoising-like effect on BG frames by eliminating temporal changes as low-pass filtering. Therefore, bits consumption for BG areas is significantly reduced. Meanwhile, occluded FG areas persistently retained in the CBG sequence require no additional coding bits. We also employ HEVC standard test sequence of Class E to simulate low-noise indoor surveillance scenarios with fewer and larger instances. In this scenario, BG compression gains are negligible due to low-noise of the test sequence. While for FG instances, the BG-FG separate encoding strategy not only avoids bitrate increase but also enhances foreground quality through appropriate block size settings that further optimizes CU size decision-making in the encoder. The rate distortion curves are depict in "Fig. 8" which demonstrate the superior performance of our proposed method.

The proposed scheme obtain small improvement on the CUHK_Avenue sequence. The sequence contains abundant FG instances but extremely sparse BG regions, making it challenging to reconstruct a high-quality CBG. Although auto-SAM2 offers significant advantages, its performance diminishes when handling numerous overlapping instances, leading to reduced segmentation precision, FG intrusion into CBG and consequently limited coding performance. Additionally, the coding efficiency gain on sequence CAVIAR_WalkByShop1front is also limited, which is due to that the scene contains almost no FG instances and is dominated by low-noise static BG.

We also compare our proposed scheme with other surveillance video coding methods encoding EWAP test sequences. All comparison methods use x265 encoder as benchmark to obtain BD-rate measure R-D performance. Experimental results shown in Table II demonstrates that our proposed scheme outperforms the method ASVC in [6] and BCBR in [4]. Moreover, our method achieves higher the BD-rate reduction compared to DCVC-DC [28] based on the the neural video codec.

TABLE I: BD-rate comparison results of the proposed method and x265.

Sequence Name	BD-rate(EWPSNR, %)
EWAP_hotel	-23.3
CUHK_Avenue	-0.6
CAVIAR_Walk1	-5.3
CAVIAR_Walk2	-25.4
CAVIAR_Walk3	-23.1
CAVIAR_WalkByShop1front	-0.3
CAVIAR_Browse1	-13.3
CAVIAR_Browse2	-21.7
CAVIAR_Browse3	-15.7
CAVIAR_Browse4	-15.1
Class_E_KristenAndSara	-10.1
Class_E_Johnny	-16.5

TABLE II: BD-rate comparison results of the proposed method and others.

Method	BD-rate(%)
BCBR [4]	-10.2
ASVC [6]	-11.2
DCVC-DC [28]	–7.7
Our method	-23.3

IV. Conclusions

In this work, we propose a surveillance video compression method based on foreground-background segmentation. By establishing an auto-SAM2 framework, it automatically identifies FG instances in original frames and classifies them as foreground (FG) frames and background (BG) frames at block level. The construction of complete BG (CBG) frames from BG frames significantly improves BG encoding efficiency. The FG sequence and CBG sequence are encoded respectively. Finally, during decoding, the FG frame is overlaid on the CBG frame to obtain reconstructed frame. Through this method, the BG bitrate is significantly reduced while FG quality remains uncompromised, and is even enhanced by the block-level segmentation mechanism. This method can be coupled with existing standard video encoders to achieve superior coding efficiency.

References

[1] H. Xu, M. Fang, L. Li, Y. Tian, and Y. Li, "The value of data mining for surveillance video in big data era," in *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, 2017, pp. 202–206. DOI: 10.1109/ICBDA.2017.8078808.

[2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the h.264/avc standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, 2007. DOI: 10.1109/TCSVT.2007.905532.

[3] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012. DOI: 10.1109/TCSVT.2012.2221191.

[4] F. Chen, H. Li, L. Li, D. Liu, and F. Wu, "Block-composed background reference for high efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 12, pp. 2639–2651, 2017. DOI: 10.1109/TCSVT.2016.2593599.

[5] X. Zhang, Y. Tian, T. Huang, S. Dong, and W. Gao, "Optimizing the hierarchical prediction and coding in hevc for surveillance and conference videos with background modeling," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4511–4526, 2014. DOI: 10.1109/TIP.2014.2352036.

[6] Y. Zhao, S. Tang, and M. Ye, "Adaptive surveillance video compression with background hyperprior," *IEEE Signal Processing Letters*, vol. 32, pp. 456–460, 2025. DOI: 10.1109/LSP.2024.3521663.

[7] V. Tiwari, D. Chaudhary, and V. Tiwari, "Foreground segmentation using gmm combined temporal differencing," in *2017 International Conference on Computer, Communications and Electronics (Comptelix)*, 2017, pp. 426–430. DOI: 10.1109/COMPTELIX.2017.8004007.

[8] S. Taneja, C. Gupta, S. Aggarwal, and V. Jindal, "Mfz-knn — a modified fuzzy based k nearest neighbor algorithm," in *2015 International Conference on Cognitive Computing and Information Processing(CCIP)*, 2015, pp. 1–5. DOI: 10.1109/CCIP.2015.7100689.

[9] X. Yang and T. Feng, "Knn non-parametric kernel density estimation method for motion foreground detection based on gaussian filtering," in *2019 11th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 2, 2019, pp. 93–96. DOI: 10.1109/IHMSC.2019.10117.

[10] P. Chhabra and S. Goyal, "An examination of the feasibility of various deep learning object detecting techniques," in *2023 International Conference on Disruptive Technologies (ICDT)*, 2023, pp. 237–242. DOI: 10.1109/ICDT57929.2023.10151099.

[11] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: State of the art," *International Journal of Multimedia Information Retrieval*, vol. 9, pp. 171–189, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:220280409.

[12] R. Lavrynenko and N. Ryabova, "Transforming semantic segmentation into instance segmentation with a guided u-net," in *2023 IEEE 18th International Conference on Computer Science and Information Technologies (CSIT)*, 2023, pp. 1–4. DOI: 10.1109/CSIT61576.2023.10324276.

[13] F. Shen, "Improvement methods for medical image segmentation based on attention u-net networks," in *2024 5th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)*, 2024, pp. 1231–1234. DOI: 10.1109/AINIT61980.2024.10581570.

[14] S. Chen, H. Yang, J. Fu, *et al.*, "U-net plus: Deep semantic segmentation for esophagus and esophageal cancer in computed tomography images," *IEEE Access*, vol. 7, pp. 82 867–82 877, 2019. DOI: 10.1109/ACCESS.2019.2923760.

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988. DOI: 10.1109/ICCV.2017.322.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017. DOI: 10.1109/TPAMI.2016.2577031.

[17] N. Ravi, V. Gabeur, Y.-T. Hu, *et al.*, "Sam 2: Segment anything in images and videos," *ArXiv*, vol. abs/2408.00714, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:271601113.

[18] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002. DOI: 10.1109/34.1000236.

[19] A. Manzanera and J. Richefeu, "A robust and computationally efficient motion detection algorithm based on sigma-delta background estimation," in *Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP'04)*, Kolkata, India, Dec. 2004. [Online]. Available: https://hal.science/hal-01222695.

[20] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 9156–9165. DOI: 10.1109/ICCV.2019.00925.

[21] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:11212020.

[22] T. Kynkäänniemi, T. Karras, S. Laine, J. Lehtinen, and T. Aila, "Improved precision and recall metric for assessing generative models," in *Neural Information Processing Systems*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:118648975.

[23] Y. Hou, Z. Ma, C. Liu, and C. C. Loy, "Learning lightweight lane detection cnns by self attention distillation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1013–1021. DOI: 10.1109/ICCV.2019.00110.

[24] C. Bettini, G. Civitarese, and R. Presotto, "Caviar: Context-driven active and incremental activity recognition," *Knowledge-Based Systems*, vol. 196, p. 105 816, 2020, ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2020.105816. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705120301969.

[25] Y. Huang, L. Song, R. Xie, Z. Luo, and X. Wang, "An mcmc based efficient parameter selection model for x265 encoder," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1–5. DOI: 10.1109/ISCAS.2018.8351034.

[26] D. Ai, J. Wang, T. He, H. Yuan, Y. Liu, and N. Ling, "Temporal and spatial perception: A novel perceptual rate-distortion optimization method for h.266/vvc encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2025. DOI: 10.1109/TCSVT.2025.3544542.

[27] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," *ITU-T VCEG-M33, April, 2001*, 2001.

[28] J. Li, B. Li, and Y. Lu, "Neural video compression with diverse contexts," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 22 616–22 626. DOI: 10.1109/CVPR52729.2023.02166.