Journal of Imaging

MDPI

Article

Recurrence Quantification Analysis for Scene Change Detection and Foreground/Background Segmentation in Videos

Theodora Kyprianidi [1], Effrosyni Doutsi [1,*] and Panagiotis Tsakalides [1,2]

[1] Foundation for Research and Technology—Hellas, 70013 Heraklion, Greece; kyptheod@ics.forth.gr (T.K.); tsakalid@ics.forth.gr (P.T.)
[2] Computer Science Department, University of Crete, 71500 Heraklion, Greece
* Correspondence: edoutsi@ics.forth.gr

Abstract: This paper presents the mathematical framework of Recurrence Quantification Analysis (RQA) for dynamic video processing, exploring its applications in two primary tasks: scene change detection and adaptive foreground/background segmentation. Originally developed for time series analysis, Recurrence Quantification Analysis (RQA) examines the recurrence of states within a dynamic system. When applied to video streams, RQA detects recurrent patterns by leveraging the temporal dynamics of video frames. This approach offers a computationally efficient and robust alternative to traditional deep learning methods, which often demand extensive training data and high computational power. Our approach is evaluated on three annotated video datasets: Autoshot, RAI, and BBC Planet Earth, where it demonstrates effectiveness in detecting abrupt scene changes, achieving results comparable to state-of-the-art techniques. We also apply RQA to foreground/background segmentation using the UCF101 and DAVIS datasets, where it accurately distinguishes between foreground motion and static background regions. Through the examination of heatmaps based on the embedding dimension and Recurrence Plots (RPs), we show that RQA provides precise segmentation, with RPs offering clearer delineation of foreground objects. Our findings indicate that RQA is a promising, flexible, and computationally efficient approach to video analysis, with potential applications across various domains requiring dynamic video processing.

Keywords: recurrence quantification analysis (RQA); dynamic video processing; scene change detection; foreground/background segmentation; video analysis

check for updates

Academic Editors: Marco La Cascia and Goorak Kwon

Received: 30 January 2025
Revised: 17 March 2025
Accepted: 31 March 2025
Published: 8 April 2025

Citation: Kyprianidi, T.; Doutsi, E.; Tsakalides, P. Recurrence Quantification Analysis for Scene Change Detection and Foreground/Background Segmentation in Videos. *J. Imaging* **2025**, *11*, 113. https://doi.org/10.3390/jimaging11040113

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Recent statistics indicate that the volume of data captured, transmitted, or stored worldwide each day is approximately 149 zettabytes, which is double the amount recorded in 2021 [1]. Furthermore, videos account for the majority of internet data traffic, representing over 50% of the total [2]. Therefore, there is an urgent need to enhance the algorithms currently employed for video analysis, understanding, and compression. Traditionally, videos are captured as a series of frames that contain significant spatial and temporal redundancy. Consequently, for a wide range of tasks, including scene change detection, video segmentation, object detection/tracking, and video compression, it is necessary to conduct extensive comparisons between consecutive frames to minimize redundancy and extract key information from the visual scene. However, this frame-based processing significantly increases computational costs, leading to energy-intensive algorithms that are inefficient, particularly for devices with limited battery life.

In the past decade, event-based technology has undergone extensive research and development to dynamically capture and process visual information in a way that mimics the neural architecture of the human eye and our perception of the world. This technology operates on the principle that crucial information is represented by rapid changes, referred to as "events", inspired by the spiking behavior of biological neurons. Event-based technology has found widespread application in sensors [3], neuromorphic hardware [4], and spiking neural networks [5], which function similarly to artificial neural networks while ensuring energy efficiency.

In this work, we present an alternative method for dynamic video stream processing using Recurrence Quantification Analysis (RQA). Originally developed for time series analysis to detect recurrent patterns, RQA leverages the concept of recurrence, which parallels redundancy in image and video processing—spatiotemporal redundant information refers to data that recurs over time. The application of RQA to video streams was first introduced in [6]. Here, we enhance the mathematical framework of RQA and explore its properties in two key applications: (i) scene change detection and (ii) adaptive foreground/background segmentation over time. Our experiments demonstrate that RQA is computationally efficient and robust, outperforming state-of-the-art deep learning methods in scene change detection. Furthermore, its application to adaptive foreground/background segmentation shows promising results, enabling motion detection approaches akin to event-based processing.

Detecting scene changes in videos is a fundamental step in applications related to visual information retrieval and scene understanding. Over time, various techniques have been developed to dynamically identify these transitions, broadly categorized into traditional methods and deep learning-based approaches. Traditional scene change detection methods primarily rely on measuring similarity between consecutive frames within a sliding window [7], handcrafted features such as color, histograms, and image gradients [8], or pixel-level comparisons, global histograms, block-based histograms, and motion-based histogram techniques [9]. More recent research has shifted towards deep learning (DL) approaches, utilizing Support Vector Machines (SVMs) [10] and Convolutional Neural Networks (CNNs) either to extract relevant features or to train models for scene change prediction. Additionally, Long Short-Term Memory (LSTM) networks have been employed to aggregate temporal information from CNN-extracted features, enabling more robust scene classification [11] and Variational Auto-Encoders (VAE) have been used to identify significant changes in the dynamic scenes of maritime video data [12].

The structure of this paper is as follows: Section 2 provides a brief overview of RQA methods, initially developed for time-series analysis and subsequently adapted for images. Section 3 presents the mathematical framework of the proposed approach, enabling the application of RQA to video streams. The experimental results are discussed in Section 4, which is divided into two parts: the first focuses on the materials, methods, and results when RQA is applied to scene change detection, while the second addresses the materials, methods, and results for foreground/background segmentation. This section also includes all relevant discussions of the experimental findings. Finally, Section 5 summarizes the conclusions of this work.

2. Recurrence Quantification Analysis: Theoretical Framework

Recurrence Quantification Analysis (RQA) is a technique for analyzing time series data by examining the recurrence properties of dynamical systems. Its primary aim is to uncover patterns and dynamics within the system, including determining the optimal embedding dimensionality D and time delay τ for reconstructing the phase-space trajectory, as proposed by Takens [13]. This reconstructed phase space allows for the identification of recurrence structures, enabling the analysis of whether the original signal exhibits recurrence.

This phase-space reconstruction captures the underlying dynamics of the system by embedding the observable time series $x = (x_1, x_2, ..., x_n)$ into a higher-dimensional space. Specifically, the reconstructed trajectory is represented as a set of vectors in the form:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_{n-(D-1)\tau} \end{bmatrix} = \begin{bmatrix} x_1 & x_{1+\tau} & \cdots & x_{1+(D-1)\tau} \\ x_2 & x_{2+\tau} & \cdots & x_{2+(D-1)\tau} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n-(D-1)\tau} & x_{n-(D-1)\tau+\tau} & \cdots & x_n, \end{bmatrix} \tag{1}$$

where $\mathbf{X}$ comprises a set of vectors $\mathbf{X_i}$ generated by starting at an initial point and selecting $D-1$ consecutive points with a time offset of τ. This technique ensures that the reconstructed dynamics preserve the original system's topological properties, allowing for the analysis of recurrence structures. By studying the recurrence patterns within this reconstructed space, one can gain insights into the system's stability, periodicity, and chaotic behavior, even with a single observable variable.

To illustrate recurrence within dynamical systems Eckmann et al. [14] introduced the Recurrence plots (RP). In the recurrence plot (RP), recurrent points are displayed in black, while non-recurrent points are shown in white. The recurrent points are defined by all pairwise distances $||\mathbf{X_i} - \mathbf{X_j}||$ between vectors $\mathbf{X_i}$ and $\mathbf{X_j}$. If the distance is smaller than a threshold ε then the point $R_{i,j}$ in RP is considered recurrent, following the formula:

$$R_{i,j} = \Theta(\varepsilon - ||\mathbf{X}_i - \mathbf{X}_j||), \tag{2}$$

where $\Theta(x)$ represents the Heaviside step function, which equals 0 for $x < 0$ and 1 for $x \geq 0$. The selection of the threshold ε is critical for identifying recurrences within the RP. Striking the right balance is essential: ε must be small enough to maintain precision while still ensuring an adequate number of recurrences and recurrent structures [15]. Since there is no universal guideline for choosing ε, its value should be tailored to the specific application and experimental conditions.

The RP is symmetric, thus the main diagonal line inherently comprises recurrent points, as $R_{i,i} = 1$ by definition. To be able to quantify the RPs, various metrics for recurrence quantification analysis have been developed [16–18]. Unclear dynamical behaviors in the original time series can be revealed by the measures of RQA. Some of these metrics are the recurrence rate (RR), which indicates the percentage of recurrent points in the RP, the Determinism (DET), which measures the recurrent points present in diagonal structures, the maximal diagonal length (Lmax) excluding the main diagonal, and the entropy (ENT), which quantifies the signal complexity in bits per bin.

RQA for Image Analysis

An extension of the one-dimensional method of the RPs and their quantification to higher dimensions was proposed by Marwan et al. [19] and later by Wallot et al. [20]. The extension of the method to higher dimensions can serve as an effective approach to manage and analyse multiple features [21]. Their method was applied in 2D images to reveal and quantify recurrent structures within images. For a d-dimensional system, they defined an n-dimensional RP using the following formula:

$$\mathbf{R_{i,j}} = \Theta(\varepsilon - ||\mathbf{X_i} - \mathbf{X_j}||), \tag{3}$$

where $\mathbf{i}$ is the d-dimensional coordinate vector and $\mathbf{X_i}$ is the phase space vector at the location given by the coordinate vector $\mathbf{i}$. In other words, the vector $\mathbf{i}$ stands for the coordinates (i_1, i_2) where i_1 and i_2 are the row and column indexes of the input image,

respectively. The dimension of the resulting RP is $n = 2 * d$, and even though it cannot be visualized, its quantification is still possible by computing the RQA measures in the n-dimensional RP. In the one-dimensional approach, the RP features a main diagonal line. Similarly, in the n-dimensional RP, there are diagonally oriented structures of d dimensions. When the method's input is a 2D image, the resulting RP is a 4D plot, and slices of that RP can be visualized as 3D subsections of the 4D RP.

The authors in [19] showed that using this extension typical spacial structures could be distinguished employing recurrences. They applied this method to biomedical images, to evaluate the bone structure from CT images of the human proximal tibia. The authors in [22] introduced a method that utilizes a fuzzy c-means clustering approach and RQA measures (FCM-RQAS) for extracting image features, which can then be employed for training a classifier.

3. Proposed Methodology

The general pipeline of this work is illustrated in Figure 1. The video stream undergoes pre-processing to generate the necessary vectors, which are then processed using RQA and compared to construct the RP. In the top-right section, we outline the key processing steps involved in scene change detection. This process relies on masks to identify frames where the scene content changes. In the bottom-right section, we illustrate the methodology for segmenting the visual scene into foreground and background. A crucial step in this process is applying the FNN algorithm, which estimates the D parameter for each patch. This parameter is then used to generate the grayscale heatmap.

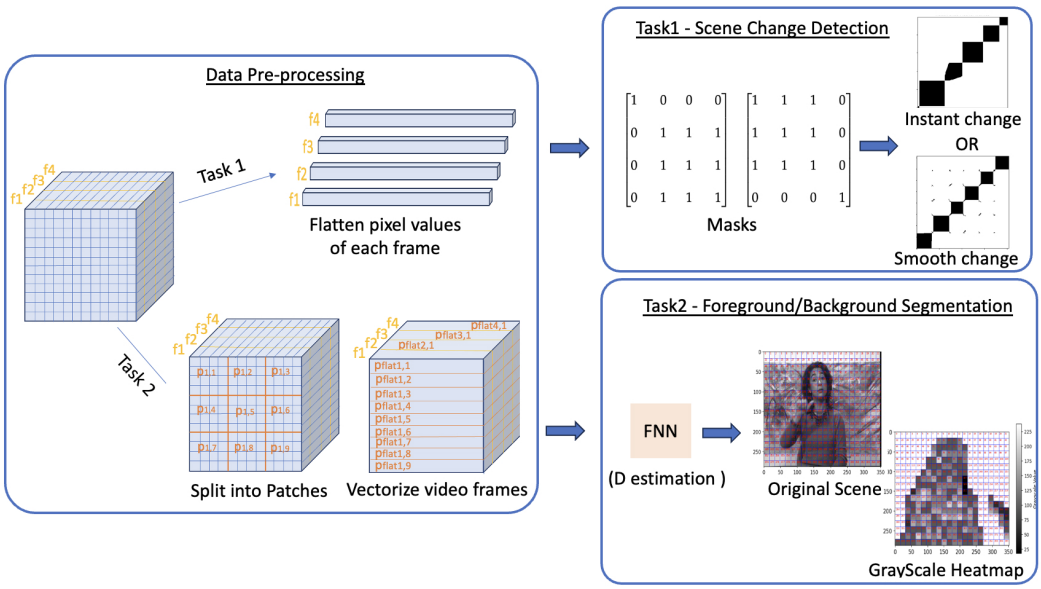


Figure 1. Proposed Methodology. The methodology consists of three distinct blocks: the first involves data pre-processing, which is essential for generating the vectors processed by RQA. The second block focuses on scene change detection, while the third block addresses foreground/background segmentation.

3.1. RQA for Foreground/Background Segmentation in Videos

In our previous work, we introduced an extension of the RQA method to handle higher-dimensional data. In this study, we build upon that initial approach by enhancing its mathematical framework and improving its representation. Consider a video stream consisting of m frames, where each frame has dimensions $M \times N$ pixels, as defined below:

$$\mathbf{f} = (\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \ldots, \mathbf{f}_m), \tag{4}$$

where $\mathbf{f}$ represents the video and $\mathbf{f}_i$ denotes the i-th frame. Each frame is divided into smaller patches, with the patch size (M_p, N_p) determined by the frame dimensions (M, N). We assume no overlap between patches, thus, within a frame of size (M, N), there are $M/M_p \times N/N_p$ non-overlapping patches. Equation (5) illustrates the first frame of the video, $\mathbf{f}_1$, divided into patches $\mathbf{p}_{i,j}$, where i denotes the frame number, and j identifies the patch within that frame:

$$\mathbf{f}_1 = \begin{bmatrix} \mathbf{p}_{1,1} & \mathbf{p}_{1,2} & \cdots & \mathbf{p}_{1,N/N_p} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{p}_{1,(M/M_p-1)*(N/N_p)+1} & \mathbf{p}_{1,(M/M_p-1)*(N/N_p)+2} & \cdots & \mathbf{p}_{1,(M/M_p)*(N/N_p)} \end{bmatrix} \tag{5}$$

For two representative patches, $\mathbf{p}_{1,1}$ and $\mathbf{p}_{1,(M/M_p)*(N/N_p)}$, corresponding to the $M_p \times N_p$ top-left and bottom-right pixels of frame $\mathbf{f}_1$, respectively, the content of these patches is given as follows:

$$\mathbf{p}_{1,1} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,N_p} \\ \vdots & \ddots & \vdots \\ x_{M_p,1} & \cdots & x_{M_p,N_p} \end{bmatrix}, \quad \mathbf{p}_{1,(M/M_p)*(N/N_p)} = \begin{bmatrix} x_{M-M_p+1,N-N_p+1} & \cdots & x_{M-M_p+1,N} \\ \vdots & \ddots & \vdots \\ x_{M,N-N_p+1} & \cdots & x_{M,N} \end{bmatrix}. \tag{6}$$

Here, $(x_{1,1}, \ldots, x_{M,N})$ represent the pixels of the frame, with coordinates $(1, 1)$ for the top-left corner and (M, N) for the bottom-right corner.

Each patch of size $M_p \times N_p$ is finally flattened resulting in a row vector of $(M_p \times N_p)$ pixel values, as shown in Equation (7)

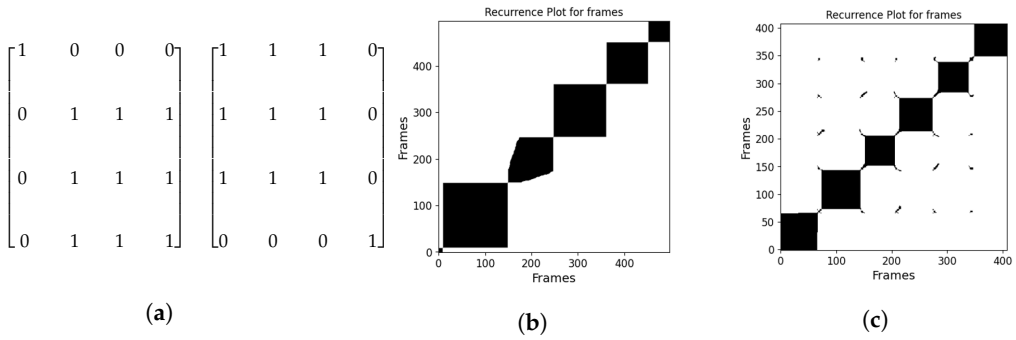
$$\begin{aligned} \mathbf{p}_{\mathbf{flat_{1,1}}} &= \left[x_{1,1}, x_{1,2}, \ldots, x_{M_p,N_p}\right] \\ \mathbf{p}_{\mathbf{flat_{1,(M/M_p)*(N/N_p)}}} &= \left[x_{M-M_p+1,N-N_p+1}, x_{M-M_p+1,N-N_p+2}, \ldots, x_{M,N}\right] \end{aligned} \tag{7}$$

Next, we have to determine the RQA parameters, i.e., the embedding dimension (D) and the time lag (τ), and then calculate vector V (Equation (8)) for each patch through the frames of the video. We obtain as many vectors V as the number of patches the video is divided, then we apply Equation (2) to obtain the RP for each patch and we observe the recurrences of each patch over time (frames).

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \\ \mathbf{V}_{m-(D-1)\tau} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \mathbf{p}_{\text{flat}_{1,1}} \\ \vdots \\ \mathbf{p}_{\text{flat}_{1+(D-1)\tau,1}} \end{bmatrix} & \begin{bmatrix} \mathbf{p}_{\text{flat}_{2,1}} \\ \vdots \\ \mathbf{p}_{\text{flat}_{2+(D-1)\tau,1}} \end{bmatrix} & \cdots & \begin{bmatrix} \mathbf{p}_{\text{flat}_{m-(D-1)\tau,1}} \\ \vdots \\ \mathbf{p}_{\text{flat}_{m,1}} \end{bmatrix} \end{bmatrix}^T \tag{8}$$

3.2. RQA for Scene Change Detection

When RQA is used for scene change detection, the video frames are not divided into smaller patches but run through whole frames. We have a resulting RP for each video. The RP is then scanned with a mask to identify the frames where a scene change occurs. We applied two masks, one to determine the starting frame of a scene and one for the ending frame of a scene. The mask shown in Figure 2a is named mask3 because it has three 1s in the square of 1s. As explained in the experimental results section, we tried the same mask but with different sizes of squares of 1s, and the mask that gave the best results was mask5 in terms of F1 score of our method compared to the ground truth for the data used.



(**a**) (**b**) (**c**)

Figure 2. (**a**) (**Left**) This mask determines the first frame of the scene. (**Right**) This mask indicates the last frame of the scene. (**b**,**c**) Recurrence Plot for whole frames each black block defines a scene, for (**b**) the scene change is instant and for (**c**) it is gradual.

One of the main parameters of RQA is the threshold ε. For scene change detection ε is set to low PSNR values, since the whole frame is analyzed and we want to detect a major difference between two frames. The masks we are using identify the starting and ending frame, by finding the edges of the black blocks along the diagonal. Figure 2b presents the RP for a video with approximately 500 frames, featuring sharp transitions between scenes. Each black block along the diagonal represents a scene, indicating that this video consists of five consecutive scenes. In contrast, Figure 2c displays the RP for another video with smooth scene transitions. Here, discontinuities appear between the black blocks representing six different scenes, resulting from the mask's structure, which struggles to accurately capture the scene changes.

3.3. Parameter Settings

3.3.1. Tuning the Threshold ε

The choice of threshold ε is crucial and depends on the application. In video data analysis, identifying recurrent patches within frames is essential. However, the Euclidean distance between vector pairs $\mathbf{V}_i, \mathbf{V}_j$ lacks significance in image processing and was ineffective for selecting ε. To address this, image quality metrics such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) were introduced to establish a meaningful ε. Instead of computing all pairwise Euclidean distances, pairwise PSNR or SSIM values are calculated and stored in a symmetric matrix. Due to the computational cost of SSIM, the PSNR-based approach was chosen, allowing for the identification of recurrent patches that satisfy the recurrence plot (RP) equation for a given $\varepsilon_{\text{PSNR}}$.

$$\mathbf{R}_{i,j} = \Theta\left(\varepsilon_{\text{PSNR}} - \frac{1}{\text{PSRN}(\mathbf{V}_i, \mathbf{V}_j)}\right), \tag{9}$$

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}(\mathbf{V}_i, \mathbf{V}_j)}\right), \tag{10}$$

where $\varepsilon_{\text{PSNR}}$ is the threshold value using the PSNR, MAX is the maximum pixel value found in the frame (i.e., for a grayscale image this value is 255), and MSE is the mean square error between the pixels of the original and the reconstructed image.

3.3.2. Selection of the Dimensionality D

The embedding dimension D is another crucial parameter in RQA. The False Nearest Neighbors (FNN) method, introduced by Kennel et al. [23], is used to determine the optimal D. This approach evaluates the change in distance between neighboring points in phase space as D increases. If embedding in a higher dimension significantly alters the distance between neighboring points, they are considered false neighbors. The optimal D is reached

when no further changes occur. We apply the FNN criterion to each patch of video frames. Starting from $D = 1$, we reconstruct phase space vectors $\mathbf{V}$ and compute pairwise PSNR distance matrices for each D. For every row in the PSNR matrix, we track the position of the maximum PSNR value and compare it across successive embedding dimensions. If the difference exceeds a threshold t_{PSNR}, the points are classified as false neighbors. The percentage of false neighbors is computed for each D, and the process is continued iteratively until stabilization is achieved.

3.3.3. Estimating the Time Delay τ

The optimal time delay τ is determined as the first minimum of the Average Mutual Information (AMI) function, averaged across all data dimensions [24]. In this study, we set the time delay parameter to $\tau = 1$, meaning that frames are processed sequentially without skipping any intermediate ones, as our primary focus is on capturing motion changes.

3.3.4. Tuning the Patch Size (M_p, N_p)

Patch size is a crucial parameter, particularly for the foreground/background segmentation approach, as it significantly affects the segmentation accuracy. If the patch size is too large, the accuracy decreases because critical regions that separate foreground motion from the static background may be lost. In contrast, setting the patch size too small improves segmentation accuracy, but comes at the cost of significantly higher computational complexity. In this work, the patch size has been set to 8×8, as it provides a good balance between accuracy and computational cost. A more detailed comparative analysis of patch size is available in the Supplementary File.

4. Experimental Results

4.1. Dataset for Scene Change Detection

The use of RQA for scene change detection is evaluated in three annotated video datasets, each indicating the frame where a scene change occurs. These 3 datasets were also used by the Autoshot approach [25], which applies neural architecture search within a space that integrates advanced 3D ConvNets and Transformers. The first dataset, Autoshot [25], comprises 853 short videos, each lasting less than one minute. The second dataset, RAI [26], includes 10 randomly selected broadcast videos, each 3–4 min long, sourced from the Rai Scuola video archive and mainly featuring documentaries and talk shows. The third dataset is the BBC Planet Earth documentary series [27], consisting of 11 long videos, each approximately 50 min long.

4.2. Results for Scene Change Detection

We applied our method to the previously mentioned datasets: Autoshot, RAI, and BBC. Due to the sharpness of the masks used in our approach, our method is more effective at detecting abrupt scene changes compared to gradual transitions. We tested various values of ε ranging from 12 to 20, finding that the optimal values for F1 score were $\varepsilon_{\text{PSNR}} = 15$ for Autoshot and RAI, and $\varepsilon_{\text{PSNR}} = 18$ for the BBC dataset. Additionally, as shown in Table 1, the best results in terms of F1 score were achieved with mask5, when comparing our method to the ground truth for the tested data.

Table 1. F1, precision, and recall scores for different mask sizes and datasets.

	RAI			Autoshot			BBC		
	F1	**PRE**	**REC**	**F1**	**PRE**	**REC**	**F1**	**PRE**	**REC**
mask9	0.823	0.961	0.719	0.750	0.789	0.714	0.778	0.982	0.645
mask7	0.832	0.948	0.741	0.762	0.790	0.737	0.781	0.981	0.648
mask5	0.835	0.939	0.752	0.757	0.750	0.765	0.783	0.972	0.655
mask3	0.829	0.912	0.760	0.747	0.709	0.790	0.776	0.935	0.662
mask2	0.588	0.475	0.719	0.529	0.393	0.809	0.547	0.464	0.664

Table 2 presents the F1 scores for the three datasets, considering both all scene changes and only abrupt scene changes. Mask5 was used for all tests, with the corresponding ε values for each dataset listed. The F1 scores for abrupt changes alone are higher across all three datasets, demonstrating that our method more accurately detects instant scene changes. Next, we compare our method with other methods that identify scene changes. Table 3 shows the F1 scores for different video scene change detection methods across various datasets. Autoshot (2023) represents a state-of-the-art approach leveraging neural networks to analyze video data. In contrast, our proposed method, RQA, applies a mathematical framework directly to the video data. We need to highlight that despite the difference in approach, RQA has a comparable F1 score with Autoshot method. Additionally, in some cases, it outperforms other methods and/or has comparable results with them. Notably, most of the methods use neural networks, and the results for RQA highlight its effectiveness and potential. One of the most important reasons to claim that is the deep learning (DL) models require distinct training processes for each specific task, relying on separate datasets, and the knowledge gained remains task-dependent. For example, while the proposed method can simultaneously perform different tasks like foreground/background segmentation and scene change detection without any training, DL models must address these tasks individually, requiring dedicated training for each.

Table 2. (**a**) RQA results all scene changes, (**b**) RQA results only for instant scene change.

(a)			
	Autoshot	**RAI**	**BBC**
$\varepsilon_{\mathbf{PSNR}}$	**15**	**15**	**15**
TP	1894	740	4100
FP	632	48	521
FN	581	244	747
PRE	0.749	0.939	0.887
REC	0.765	0.752	0.846
F1	0.757	0.835	0.866
(b)			
	Autoshot	**RAI**	**BBC**
$\varepsilon_{\mathbf{PSNR}}$	**15**	**15**	**18**
TP	1748	782	4043
FP	632	48	521
FN	350	38	668
PRE	0.734	0.934	0.885
REC	0.833	0.947	0.858
F1	0.781	0.941	0.872

Table 3. F1 scores for different methods. Autoshot [25], DSMs [28], ST ConvNets [29], TransNet [30], TransNetV2 [31], Hierarchical clustering [32], Deep Siamese Network [33].

Method	Autoshot	RAI	BBC
Autoshot (2023)	0.841	0.971	0.955
DSMs (2018)		0.893	0.939
ST ConvNets (2017)		0.926	0.939
TransNet (2019)		0.929	0.943
TransNetV2 (2024)		0.962	0.939
RQA	0.781	0.941	0.872
Hierarchical clustering (2015)		0.720	
Deep Siamese Network (2015)			0.620

4.3. Dataset for Adaptive Foreground/Background Segmentation

This application requires short videos that feature a single scene, ideally with no simultaneous motion in the foreground and background. This means that all parts of the frame have motion, making it difficult for our method to distinguish between the two categories. UCF101 dataset [34] is a large dataset of human action, consisting of 101 action classes and more than 13k short clips, making it a very suitable video collection to evaluate our method. DAVIS dataset (Densely Annotated VIdeo Segmentation) [35], consists of 50 high-quality full HD video sequences that cover common video object segmentation challenges such as occlusions, motion blur, and appearance changes. The videos used in this work and the result can be found in the following url https://github.com/dwrakyp/MDPI_videos_results.git (accessed on 30 January 2025).

4.4. Results for Adaptive Foreground/Background Segmentation

We analyzed several videos from the UCF101 dataset [34] and the DAVIS dataset [35], selecting three videos from the UCF101 dataset to showcase the results of our RQA analysis. The first video, titled 'make-up' and shown in Figure 3 (top), features a stable camera capturing a girl applying make-up with a stationary background and motion limited to the foreground. The second video, 'parade', illustrated in Figure 3 (middle), also uses a stable camera, focusing on a road where a parade is passing by, characterized by strong foreground motion (the parade) and minimal background motion. The third video, 'ball', represented in Figure 3 (bottom), depicts a stable camera view of a natural landscape with a person on the right throwing a ball; aside from minor leaf movement in the trees, the landscape remains static. These videos were chosen to demonstrate our method's capability to handle varying types of motion: no movement, subtle movement (leaves), and significant movement (parade, make-up application).

Each video is segmented into patches, and for each patch, the optimal embedding dimension D is determined using the FNN algorithm, with $\varepsilon_{\text{PSNR}} = 35$, a value chosen experimentally. A grayscale heatmap illustrating the optimal D is displayed in the middle column of Figure 3. As anticipated, RQA effectively detects and highlights regions of motion in the image, categorizing them as foreground, while areas with minimal or no motion are classified as background. Furthermore, we investigated the role of Recurrence Plots (RPs), which convey critical insights derived from RQA beyond its standard features. The right column of Figure 3 presents a grayscale heatmap generated from the RP for each patch, where high-intensity values indicate background regions and low-intensity values represent foreground regions. Both heatmaps demonstrate effective foreground/background segmentation, but the RP-based heatmap is significantly more precise than the D-value heatmap, providing a more detailed representation of object shapes within the scene. Con-

sequently, depending on the precision required for a given application, our method offers flexibility in achieving varying levels of segmentation accuracy.

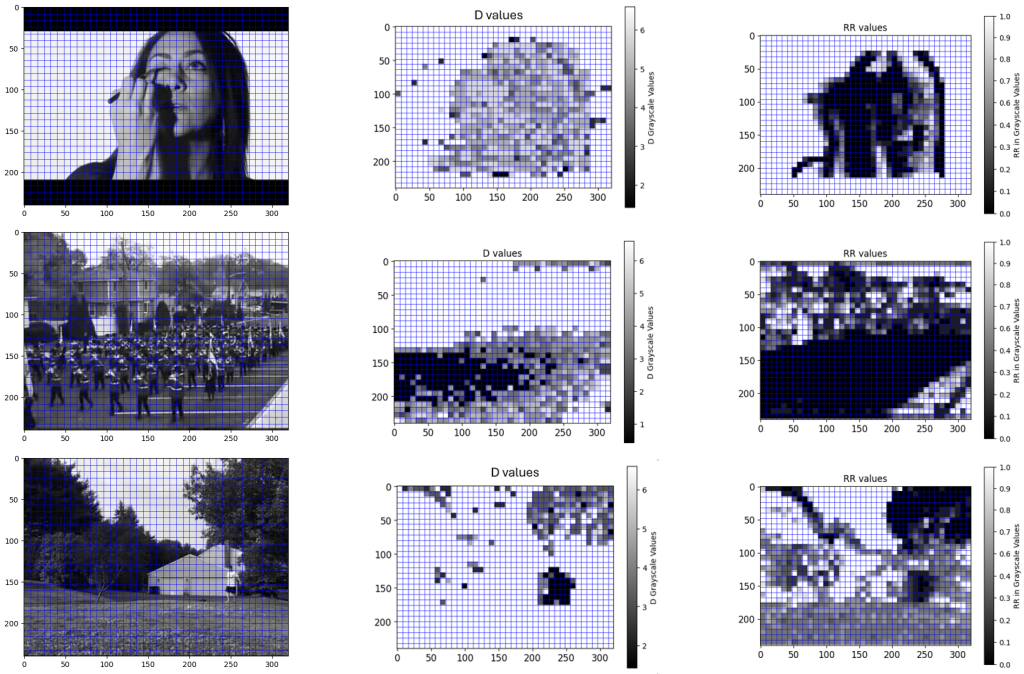


Figure 3. Results on the makeup video (**top** row), the parade video (**middle** row), and the ball video (**bottom** row). (**left** column) Random frame which is split into patches of a size 8×8. (**middle** column) Grayscale heatmap generated for the D values. (**right** column) Grayscale heatmap based on the RR values.

5. Conclusions

In this work, we have evaluated the use of Recurrence Quantification Analysis (RQA) for video analysis, focusing on two tasks: scene change detection and adaptive foreground/background segmentation.

For scene change detection, we applied RQA to three video datasets: Autoshot, RAI, and BBC Planet Earth, comparing its performance with several state-of-the-art methods. Our experiments demonstrated that RQA achieves competitive results, particularly in detecting abrupt scene changes. We showed that with optimal parameter tuning, our method can effectively detect scene changes with high F1 scores, which are comparable to the neural network-based Autoshot method. Moreover, RQA performs robustly across different datasets, indicating its generalizability and potential for practical applications.

For adaptive foreground/background segmentation, we used videos from the UCF101 and DAVIS datasets to test our method's ability to distinguish between foreground and background regions based on motion. We found that RQA successfully identifies regions with significant motion, such as the foreground, while static areas are classified as background. By analyzing the heatmaps of the embedding dimension D and Recurrence Plots (RPs), we demonstrated that our method can produce precise segmentation maps, with RPs offering clearer foreground/background distinctions. The method's flexibility allows it to adjust to different levels of precision depending on the specific needs of the application.

Overall, RQA proves to be a promising approach for both scene change detection and adaptive foreground/background segmentation, offering high accuracy and flexibility while providing insights into the dynamics of video data. Future work can explore further refinements and the integration of RQA with deep learning-based methods to enhance its performance in more complex video analysis tasks.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/jimaging11040113/s1, Figure S1: Results on the makeup video (top row), the parade video (middle row), and the ball video (bottom row). (left column) Random frame which is split into patches of a size 16 × 16. (middle column) Grayscale heatmap generated for the D values. (right column) Grayscale heatmap based on the RR values. Figure S2: Results on the makeup video (top row), the parade video (middle row), and the ball video (bottom row). (left column) Random frame which is split into patches of a size 32 × 32. (middle column) Grayscale heatmap generated for the D values. (right column) Grayscale heatmap based on the RR values. Figure S3: Results on the makeup video (top row), the parade video (middle row), and the ball video (bottom row). (left column) Random frame which is split into patches of a size 8 × 8 and for PSNR threshold $\varepsilon = 45$. (middle column) Grayscale heatmap generated for the D values. (right column) Grayscale heatmap based on the RR values. Figure S4: Results on the makeup video (top row), the parade video (middle row), and the ball video (bottom row). (left column) Random frame which is split into patches of a size 8 × 8 and for PSNR threshold $\varepsilon = 20$. (middle column) Grayscale heatmap generated for the D values. (right column) Grayscale heatmap based on the RR values. Table S1: Execution time for FNN and RQA for different patch sizes.

Author Contributions: Conceptualization, E.D.; methodology, T.K. and E.D.; software, T.K.; validation, T.K.; formal analysis, T.K. and E.D.; investigation, T.K.; writing—original draft preparation, T.K. and E.D.; writing—review and editing, P.T.; visualization, T.K.; supervision, P.T.; funding acquisition, E.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT) under grant agreement No. 330 (BrainSIM).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This study does not generate new data; all datasets utilized are publicly available and are properly cited in the main text. The code of this work is available at the following https://github.com/dwrakyp/MDPI_videos_results.git, accessed on 30 January 2025.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. 53 Important Statistics About How Much Data Is Created Every Day in 2024. Available online: https://financesonline.com/how-much-data-is-created-every-day/ (accessed on 24 December 2024).
2. How Much Data is Generated Every Day (2024). Available online: https://whatsthebigdata.com/data-generated-every-day/ (accessed on 14 May 2024).
3. Chakravarthi, B.; Verma, A.A.; Daniilidis, K.; Fermuller, C.; Yang, Y. Recent Event Camera Innovations: A Survey. *arXiv* **2024**, arXiv:cs.CV/2408.13627.
4. Gonzalez, H.A.; Huang, J.; Kelber, F.; Nazeer, K.K.; Langer, T.; Liu, C.; Lohrmann, M.; Rostami, A.; Schöne, M.; Vogginger, B.; et al. SpiNNaker2: A Large-Scale Neuromorphic System for Event-Based and Asynchronous Machine Learning. *arXiv* **2024**, arXiv:cs.ET/2401.04491.
5. Kundu, S.; Zhu, R.J.; Jaiswal, A.; Beerel, P.A. Recent Advances in Scalable Energy-Efficient and Trustworthy Spiking Neural Networks: From Algorithms to Technology. In Proceedings of the ICASSP 2024—2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; pp. 13256–13260. [CrossRef]
6. Kyprianidi, T.; Doutsi, E.; Tzagkarakis, G.; Tsakalides, P. Exploring the Potential of Recurrence Quantification Analysis for Video Analysis and Motion Detection. In Proceedings of the 2024 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 27–30 October 2024; pp. 2606–2612. [CrossRef]
7. Kowdle, A.; Chen, T. Learning to segment a video to clips based on scene and camera motion. In *Proceedings of the 12th European Conference on Computer Vision—Volume Part III*; ECCV'12; Springer: Berlin/Heidelberg, Germany, 2012; pp. 272–286. [CrossRef]
8. Salih, Y.; George, L.E. Dynamic Scene Change Detection in Video Coding. *Int. J. Eng.* **2020**, *33*, 966–974. [CrossRef]
9. Rascioni, G.; Spinsante, S.; Gambi, E. An Optimized Dynamic Scene Change Detection Algorithm for H.264/AVC Encoded Video Sequences. *Int. J. Digit. Multim. Broadcast.* **2010**, *2010*, 864123:1–864123:9.

10. Gangopadhyay, A.; Tripathi, S.M.; Jindal, I.; Raman, S. SA-CNN: Dynamic Scene Classification using Convolutional Neural Networks. *arXiv* **2015**, arXiv:cs.CV/1502.05243.
11. Peng, X.; Bouzerdoum, A.; Phung, S.L. A Trajectory-Based Method for Dynamic Scene Recognition. *Int. J. Pattern Recognit. Artif. Intell.* **2021**, *35*, 2150029. [CrossRef]
12. Trinh, L.; Anwar, A.; Mercelis, S. SeaDSC: A video-based unsupervised method for dynamic scene change detection in unmanned surface vehicles. *arXiv* **2023**, arXiv:cs.CV/2311.11580.
13. Takens, F. *Dynamical Systems and Turbulence, Warwick 1980*; Springer: Berlin/Heidelberg, Germany, 1981; pp. 366–381.
14. Eckmann, J.P.; Kamphorst, S.O.; Ruelle, D. Recurrence plots of dynamical systems. *World Sci. Ser. Nonlinear Sci. Ser. A* **1995**, *16*, 441–446.
15. Marwan, N.; Webber, C.L., Jr. Mathematical and computational foundations of recurrence quantifications. In *Recurrence Quantification Analysis: Theory and Best Practices*; Springer: Cham, Switzerland, 2014; pp. 3–43.
16. Zbilut, J.P.; Webber, C.L., Jr. Embeddings and delays as derived from quantification of recurrence plots. *Phys. Lett. A* **1992**, *171*, 199–203.
17. Webber, C.L., Jr.; Zbilut, J.P. Dynamical assessment of physiological systems and states using recurrence plot strategies. *J. Appl. Physiol.* **1994**, *76*, 965–973. [PubMed]
18. Marwan, N.; Wessel, N.; Meyerfeldt, U.; Schirdewan, A.; Kurths, J. Recurrence-plot-based measures of complexity and their application to heart-rate-variability data. *Phys. Rev. E* **2002**, *66*, 026702. [CrossRef] [PubMed]
19. Marwan, N.; Kurths, J.; Saparin, P. Generalised recurrence plot analysis for spatial data. *Phys. Lett. A* **2007**, *360*, 545–551. [CrossRef]
20. Wallot, S.; Roepstorff, A.; Mønster, D. Multidimensional Recurrence Quantification Analysis (MdRQA) for the analysis of multidimensional time-series: A software implementation in MATLAB and its application to group-level data in joint action. *Front. Psychol.* **2016**, *7*, 1835. [CrossRef] [PubMed]
21. Zervou, M.A.; Tzagkarakis, G.; Tsakalides, P. Automated screening of dyslexia via dynamical recurrence analysis of wearable sensor data. In Proceedings of the 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE), Athens, Greece, 28–30 October 2019; pp. 770–774.
22. Chomiak, T. Recurrence quantification analysis statistics for image feature extraction and classification. *Data-Enabled Discov. Appl.* **2020**, *4*, 1–9.
23. Kennel, M.B.; Brown, R.; Abarbanel, H.D. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. A* **1992**, *45*, 3403. [CrossRef] [PubMed]
24. Vlachos, I.; Kugiumtzis, D. State Space Reconstruction from Multiple Time Series. In *Topics on Chaotic Systems*; World Scientific Publishing: Singapore, 2009; pp. 378–387. [CrossRef]
25. Zhu, W.; Huang, Y.; Xie, X.; Liu, W.; Deng, J.; Zhang, D.; Wang, Z.; Liu, J. Autoshot: A short video dataset and state-of-the-art shot boundary detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 2238–2247.
26. Available online: http://imagelab.ing.unimore.it (accessed on 30 January 2025).
27. Available online: https://www.bbc.co.uk/programmes/b006mywy (accessed on 30 January 2025).
28. Tang, S.; Feng, L.; Kuang, Z.; Chen, Y.; Zhang, W. Fast video shot transition localization with deep structured models. In *Proceedings of the Asian Conference on Computer Vision*; Springer: Cham, Switzerland, 2018; pp. 577–592.
29. Hassanien, A.; Elgharib, M.; Selim, A.; Bae, S.H.; Hefeeda, M.; Matusik, W. Large-scale, fast and accurate shot boundary detection through spatio-temporal convolutional neural networks. *arXiv* **2017**, arXiv:1705.03281.
30. Lokoč, J.; Kovalčík, G.; Souček, T.; Moravec, J.; Čech, P. A framework for effective known-item search in video. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 1777–1785.
31. Soucek, T.; Lokoc, J. Transnet v2: An effective deep network architecture for fast shot transition detection. In Proceedings of the 32nd ACM International Conference on Multimedia, Melbourne, VIC, Australia, 28 October–1 November 2024; pp. 11218–11221.
32. Baraldi, L.; Grana, C.; Cucchiara, R. Shot and scene detection via hierarchical clustering for re-using broadcast video. In Proceedings of the Computer Analysis of Images and Patterns: 16th International Conference, CAIP 2015, Valletta, Malta, 2–4 September 2015; Proceedings, Part I 16; Springer: Cham, Switzerland, 2015; pp. 801–811.
33. Baraldi, L.; Grana, C.; Cucchiara, R. A deep siamese network for scene detection in broadcast videos. In Proceedings of the 23rd ACM international conference on Multimedia, Brisbane, Australia, 26–30 October 2015; pp. 1199–1202.

34. Soomro, K.; Roshan Zamir, A.; Shah, M. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. In *Technical Report CRCV-TR-12-01*; Center for Research and Computer Vision (CRCV) at University of Central Florida: Orlando, FL, USA, 2012.
35. Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Gool, L.V.; Gross, M.; Sorkine-Hornung, A. A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation. In Proceedings of the The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.