

# Rough Video Conceptualization for Real-time Event Precognition with Motion Entropy

Debarati B. Chakraborty<sup>1</sup>

*Indian Institute of Technology, Jodhpur, India*

Sankar K. Pal<sup>2</sup>

*Indian Statistical Institute, Kolkata, India*

---

## Abstract

This article defines a new methodology for pre-recognition of events with object motion analysis in a video without any prior knowledge. This unsupervised application is named as 'conceptualization'. This conceptualization technique is also tested with real-time video data in an internet of things (IoT) architecture. The merits of rough sets in the framework of granular computing are explored to execute the task. The proposed method is designed for the video sequences that are acquired by simple static RGB sensors. Here the video sequences are granulated with our newly defined 'motion granules' and then those are modeled as rough sets over this granulation for moving object/ background estimation. Video conceptualization is performed afterwards by quantifying the approximation with a new measure, namely, motion entropy. The values obtained by this measure reflect the amount of uncertainty present in the motion of each individual moving object which enables precognition of events. The effectiveness of the proposed method is verified with extensive experiments in identifying the different motion patterns present in a video sequence. The frames with possibilities of events present therein are identified with this analysis. Both offline and real-time sequences are used for this verification. An IoT architecture is formed to test the proposed algorithm with physical devices in identifying the

---

<sup>1</sup>Email: debarati.earth@gmail.com

<sup>2</sup>Email: sankar@isical.ac.in

frames containing possible events.

*Keywords:* Real-time video analysis, rough sets, granular computing, neighborhood granules, entropy, internet of things (IoT), unsupervised event precognition

---

## 1. Introduction

Video understanding and recognition of events present in it is one of the basic steps in vision-based automation. There are several types of video analysis techniques aimed towards it. Video scene understanding, behavior understanding, storyline understanding, detection of events, prediction of events are a few applications of visual automation among others. This article proposes methods to analyze the nature of different types of motions present in a video sequence and to predict possible events with detection of unpredictable change in motion of objects with rough set theoretic granular computing. This is an unsupervised approach, that is the detection is performed without any prior knowledge. This process is named as '*conceptualization*'.

Video conceptualization, in other words, is a technique of pre-recognition of events in videos with object motion analysis. That is, whether there is a probability of some events to take place and further analysis of the content of the video is required or not can be decided with video conceptualization. This process is much simpler, requires no labelled data and is faster than video understanding techniques. Therefore labelled data dependency and time consumption of an automated computer vision system (e.g. any surveillance system) can be reduced with this technique.

Granulation is a natural process of interpretation in human mind. This concept was first introduced to machine learning by Zadeh [1], with crisp granules. These information granules are used effectively in several areas of machine learning [2, 3]. Here, a new way of granule formation is defined over video sequences in two layers. The first layer of forming granules contains the spatio-color information of each frame. The second layer contains the motion information of

those spatio-color granules. That is why these are named as 'motion granules'. The object-background rough set is defined over **these** motion granules.

Theory of rough sets, as explained by Pawlak [4], has become a popular mathematical framework for granular computing. The focus of the theory is on the ambiguity caused by limited discernibility of objects in the domain of discourse. Its key concepts are those of object 'indiscernibility' and 'set approximation'. The sets defined in this work over the newly defined motion granules, inherit the overlapping property which helps to decrease the uncertainties in decision making by pushing uncertain granules towards the boundary of the sets. A new measure, namely, motion entropy is introduced **in the conceptualization method** to quantify the uncertainty present in the motion of an object. **The entropy** values can reflect well whether there is any sudden change in the motion of the object, and can differentiate between continuous and random movements. **The effectiveness of the concept of 'conceptualization' is tested with real-time video acquisition in an IoT-set up built with a Raspberry Pi 3B+ in lab environment.** To our knowledge, there does not exist any literature where the unsupervised event precognition is done from videos captured with a single static RGB camera. The methods of detection of unusual movements in IoT with real-time video acquisition is also very scarce.

The novelties that lie in this article are mainly in: i) approximating object-background with the concept of rough sets in an unsupervised manner over the newly defined 'two-layered motion granules', ii) formulating a new uncertainty measure, motion entropy to compute the uncertainty present in the movement of an object and iii) conceptualizing the videos with a simple RGB (if available) sensor, and iv) testing the effectiveness of the proposed algorithm in an IoT even with real-time data acquisition.

The rest of the article is organized as follows. A few benchmark work on video understanding and event detection are discussed in Section 2. The proposed work is explained in details in Section 3 **along with an IoT architecture of its realization/ demonstration.** This includes the formulation of two-layered motion granules, definition of object-background as rough sets over this gran-

ulation, and the definition of motion entropy. The key steps and proposed algorithms for video conceptualization are described in Section 4. Experimental results on several video sequences along with those in IoT setup are explained in  
60 Section 5 to demonstrate the effectiveness of the proposed method. The overall conclusion is drawn in Section 6.

## 2. Related Work

There exist a significant amount of work those address the task of video understanding [5]. This problem was first dealt by Brand [6] where videos of ma-  
65 nipulation tasks were interpreted with psychologically-based causal constraints to detect meaningful changes in motions. An approach of activity pattern analysis was then developed by Stauffer *et al.* [7] with accumulation of information from multiple cameras. A method of automatic discovery of the key patterns of motion was formulated by Yang *et al.* [8] by using a low level feature, pixel-wise  
70 optical flow, several of which were embedded later in a diffusion map framework. Behavior analysis with video understanding by tracking people and analyzing the trajectory with Mean-shift algorithm was formulated by Zaidenberg *et al.* [9]. An unsupervised method for video understanding was proposed by Milbich *et al.* [10] where combinatorial sequence matching was performed to train a  
75 CNN, and thereby using a huge amount of labelled data for training the CNN. Mademlis *et al.* [11] came up with a method of unsupervised video summarization with salient features. Another way of video understanding with desktop action recognition from ego-centric videos was developed by Cai *et al.* [12]. It was mainly focused on hand motion analysis. Another problem related to video  
80 understanding is re-identification of person in surveillance system. It was recently addressed by Gao *et al.* [13] with pose-guided spatio-temporal alignment.

Event recognition from videos is a step ahead inference of video understanding. There are several methods addressing this task. We are going to discuss here a few popular ones. Ke *et al.* [14] developed a method of event under-  
85 standing from crowd by matching spatio-temporal segments among consecutive

frames. The abnormal event detection and generation of description in a human-understandable format with a hybridized CNN model was developed by Himani *et al.* [15]. Rare event detection from satellite images with fine-tuned representation learning was developed by Hamaguchi *et al.* [16]. Human behavior  
90 recognition from multiview camera was **described** by Hsueh *et al.* [17] with a deep network that was developed by combining convolutional neural network and long-short-term memory network. Pre-recognition or prediction of future events, **on the other hand**, is a **more challenging** task that is addressed recently. Liang *et al.* [18] recently developed a technique of future event prediction by  
95 combining person behavior and person interaction together

All of the aforementioned approaches either need initial manual intervention or huge amount of labeled dataset for training. Therefore, the methods discussed above require different training data to be applied for different type of videos. In this work we concentrated on precognition of event(s) from gen-  
100 eral videos. Please note that this is an unsupervised technique and the entire method does not go through any rigorous training. Therefore the proposed method becomes more general method which could be applied over different type of sequences without any new information. It is different from the existing methods in methodology-wise, as well as objective-wise. In the present inves-  
105 tigation our objectives are: to indicate the possibilities of events to occur, to identify the frames with maximum information both with offline sequences and real-time sequences in IoT, and to select a few sets of consecutive frames as the highlight of the entire sequence. An IoT architecture is formed afterward to check the effectiveness of the algorithm in such a network with real-time video  
110 acquisition. Here the network is formed by embedding a computer, Raspberry pi 3B+, a Raspberry pi cam, and a bread board with a LED. The LED glows when possible event is detected in the real-time sequence.

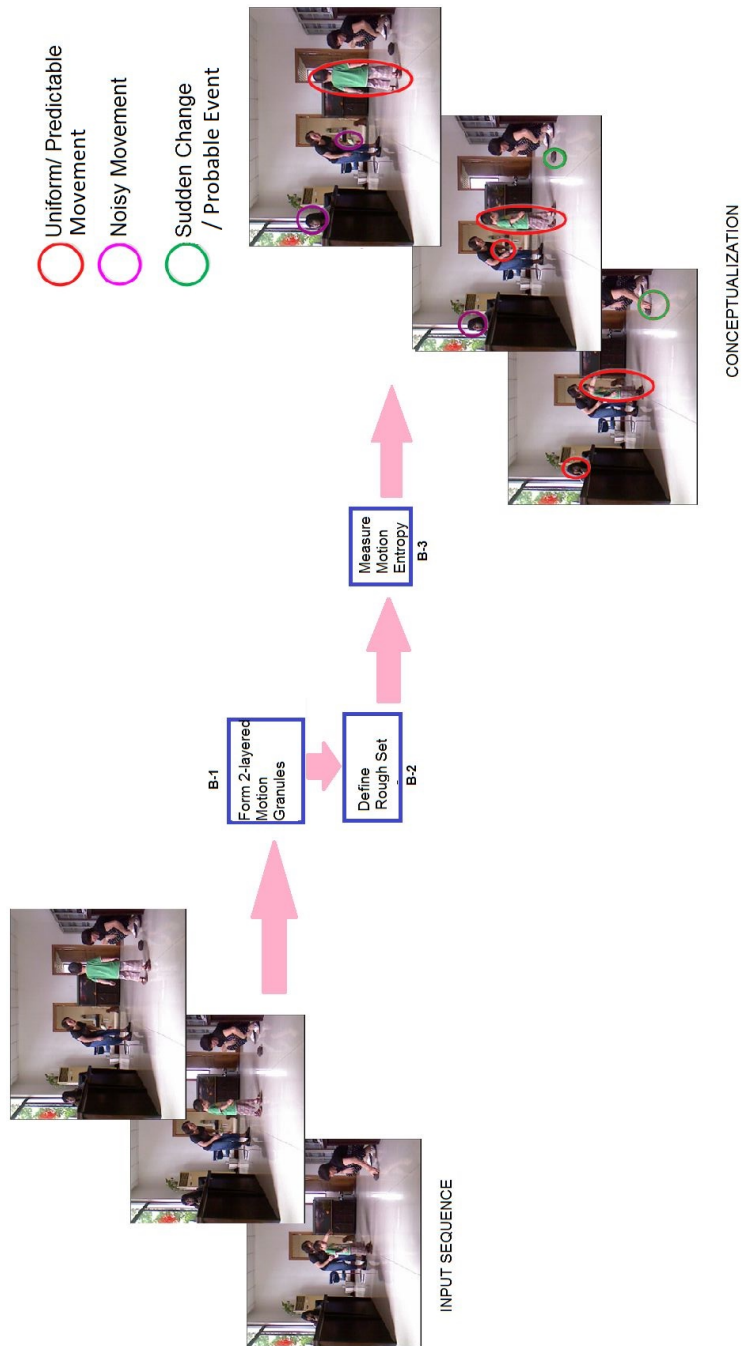


Figure 1: Key steps of video conceptualization

### 3. Proposed Work

Here an unsupervised *rough set based approach* is proposed to conceptualize the video based on the nature of moving-static elements present in it and thereby infer the possibility of some future event to take place in the scene. The moving object(s) and background are represented as rough sets over temporal domain. The granules are formed here in two layers. In the first layer the granules are formed as described in our earlier work [19]. In the second layer the motion granules are formed over the spatio-temporal granules in the temporal domain. The object and background sets are approximated as rough sets over this motion granules. The decision about the continuous moving, random moving, and sudden change (possible event) in a sequence are taken based on the nature of the neighborhood granules with respect to the sets. The nature of movements of the objects is then quantified with the newly defined motion entropy ( $M_E$ ). The regions having high  $M_E$  values are detected as the regions with more information (where some unpredictable change occurs).

The basic operational principles of this conceptualization method are shown in Figure 1. Two layered motion granules are formed over the input video sequence in the first block (B-1). Rough set is defined over this granulation in the second block (B-2). Motion entropy is then measured for the granules present in the set in the third block (B-3). The output of this method is the video sequence with the moving object(s) classified according to the nature of their movements. The working principle of B-1 is described in our earlier work [19]. Operations of blocks B-2 and B-3 are described in sections 3.2 and 3.3 respectively. Here we assumed that if there is a possibility of some event to take place there, then there would be some abrupt change in motion in a video scene. Therefore, the frames identified with sudden change may be labeled as the cause of possible future events.

**IoT Architecture:** The newly defined concept of conceptualization is tested with real-time video acquisition in IoT architecture. This architecture is shown in Fig. 2. It can be seen from Fig. 2 that real-time sequences are acquired

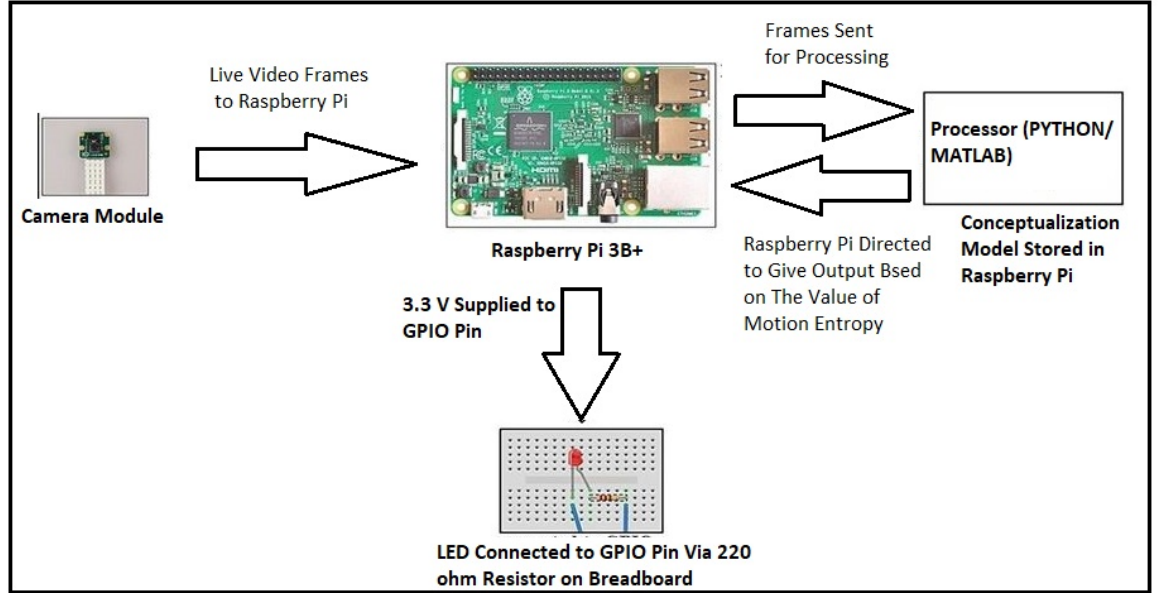


Figure 2: Proposed IoT Architecture for Video Conceptualization

with raspberry pi cam, and they are given as input to raspberry pi 3B+ mother board. The sequence is then analyzed with the proposed algorithm which is stored in raspberry pi. A bread board with a LED is connected in the output of raspberry pi. The LED glows if any unusual motion is detected or possibility of some event to occur arises in the input sequence. Details of the experiments performed with this architectural set up to demonstrate the significance of 'conceptualization' are provided in Section 5.5

### 3.1. Formation of Two-layered Motion Granules

Moving object(s) and background sets (upper and lower approximations) are defined using Pawlak's rough set (PaRS)[4] over neighborhood granules. These granules are constructed over temporal difference values of video frames. Spatio-temporal granules are formed in the first layer, and motion granules are formed in the second layer over these spatio-temporal granules. These layers are elaborated in the following sections.



### 3.1.1. 1<sup>st</sup> Layer: Spatio-temporal Granules

The formation of spatio-temporal granules is described in details in our earlier work [19]. Here we describe it in brief for the convenience of readers. Let the current frame of a sequence be denoted as  $f_t$  (size  $M \times N$ ) and its previous  $P$ -frames be  $f_{t-p} : p = 1 \dots P$ . The difference values are stored in  $P$  number of different matrices ( $T\_V_p$ ). The differences between  $f_t$  to all of its previous  $1, \dots, P$  frames are computed. It is shown in Eqn. (1).  $T\_V$  is of size  $M \times N$ .

$$T\_V_p = |f_t - f_{t-p}| \quad : \quad p = 1, \dots, P. \quad (1)$$

The spatio-temporal granules are formed considering values of the points as  $T\_V_p$  over the spatial domain. Let  $x_i$  be the position of a pixel in the difference frame ( $T\_V_p$ ), then the granule around it at  $p^{th}$  frame, denoted as  $\aleph_{sp-tmp_p}(x_i)$ , is formed according to Eqn (2).

$$\aleph_{sp-tmp_p}(x_i) = \bigcup x_j \in T\_V_p \quad (2)$$

where  $x_i$  and  $x_j$  are binary connected over  $|T\_V_p(x_i) - T\_V_p(x_j)| < Thr_t$  and  $x_j \in T\_V_p$ .

### 3.1.2. 2<sup>nd</sup> Layer: Motion Granules

The motion granules are defined to unify the similar spatio-temporal granules present in each difference frame  $T\_V_p$ . That is,  $P$  number of  $\aleph_{sp-tmp}(x_{ip})$  can be present there in each motion granule. Here  $x_{ip}$  is the location of the representative point of the spatio-temporal granule at frame  $T\_V_p$ . The motion granules will contain the optical flow information of each  $\aleph_{sp-tmp}(x_{i1})$ .

One may note that  $\aleph_{sp-tmp}(x_{ip})$  and  $\aleph_{sp-tmp}(x_{i(p-1)})$  represent two regions in two difference frames  $T\_V_p$  and  $T\_V_{p-1}$  respectively. Now, if we plot these two regions in a same frame those regions may have non-zero intersection, or a region may be a subset to another region. We have considered this phenomenon while defining the motion granules. A motion granule  $\mathcal{M}(x_i)$  over the  $P$  frames in the sequence is defined in Eqn. (3).

An example of forming motion granules over a sequence as defined in Eqn. (3) is shown in Fig. 3. Here the granules formed over moving objects are only shown for the sake of simplicity. In the first part of this figure (in the left most side) the frames in D- feature space are shown. Absolute difference is taken from frame  $f_t$  to frames  $f_{t-1}$ ,  $f_{t-2}$  and  $f_{t-3}$ . Three difference frames are obtained in this way and spatio-temporal granules are formed on the three difference frames. The three binary images ( $T\_V_1, \dots, T\_V_3$ ) in the right most side of Fig. 3 show these spatio-temporal granules. The motion granules are formed over these spatio-temporal granules as shown there in the lower part of Fig. 3. Here two motion granules (represented with red and green dotted bounding box) are formed as two moving objects are present. The red dots and green dots are the representative points in each frame with which the motion granules are formed. From this figure one can estimate how motion granules are formed taking into account the optical flow of spatio-temporal granules.

$$\mathcal{M}(x_i) = \left\{ \bigcup \mathbb{N}_{sp-tmp_p}(x_{ip}) \right\} \quad \text{if} \quad \mathbb{N}_{sp-tmp}(x_{i1}) \not\subset \mathbb{N}_{sp-tmp}(x_{ip}) \quad \text{and} \quad \mathbb{N}_{sp-tmp}(x_{i(p-1)}) \cap \mathbb{N}_{sp-tmp}(x_{ip}) \neq \emptyset \quad \forall p = 2, \dots, P. \quad (3)$$

**Note** that in Eqn. (3)  $\mathbb{N}_{sp-tmp}(x_{ip})$  denotes the spatio-temporal granule formed in the  $p^{th}$  frame. For example, the white segments in the frames  $T\_V_1$ ,  $T\_V_2$  and  $T\_V_3$  in Fig. 3 denote  $\mathbb{N}_{sp-tmp_1}(x_{i1})$ ,  $\mathbb{N}_{sp-tmp_2}(x_{i2})$  and  $\mathbb{N}_{sp-tmp_3}(x_{i3})$  for the example frames. Two motion granules, formed over these spatio-temporal granules according to Eqn. (3), are shown with red and green dotted rectangles in Fig. 3.

### 3.2. Object-Background as Rough Sets

Here the moving object(s) and background **are** defined as a rough set over the aforesaid motion granules in the current frame  $f_t$ . The information from the set of its previous  $P$  number of frames are taken into account and represented as  $\{P\} = \{t-1, \dots, t-P\}$ . The feature of a granule  $\mathcal{M}(x_i)$  that is considered here to define the sets is the signed Manhattan Distance (along  $x$  and  $y$  axes) between the consecutive points of  $\mathcal{M}(x_i)$ . This distance is denoted as  $\vec{d}_p$ . One

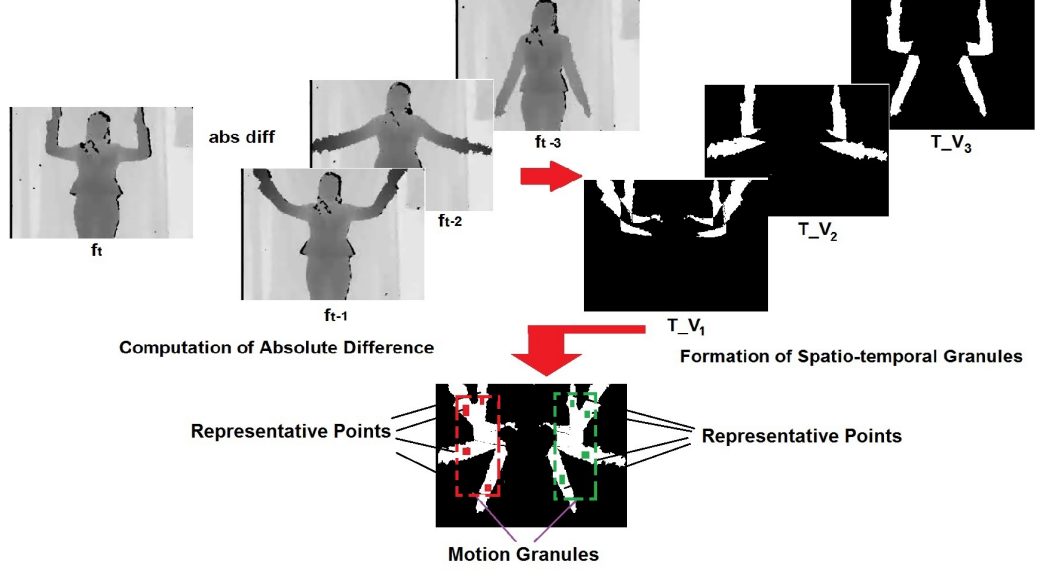


Figure 3: Example Formation of Motion Granules

may note that  $\vec{d}_p$  is a vector with four components, viz., two absolute values  
 210 (along  $x$  and  $y$  axes) and two signs (positive or negative in those axes). It is  
 computed according to Eqn. (4)

$$\vec{d}_p = x_{i(p-1)} - x_{ip}, \forall p = 2, \dots, P. \quad (4)$$

Fig. 4 represents the computation of  $\vec{d}_p$ s over two motion granules  $\mathcal{M}(x_1)$   
 and  $\mathcal{M}(x_2)$  pictorially. Here the dotted blue lines with directions signify the  
 distances between the consecutive points in a granule. That is, the signed  
 215 distance along both the  $x$  and  $y$  axes are computed here. One may note that  
 all the  $d_1$ ,  $d_2$  and  $d_3$  contain two magnitudes and two directions (positive or  
 negative) in both the granules.

A granule  $\mathcal{M}(x_i)$  is said to belong to lower approximation of the object  
 region, if the absolute values of all  $|d_p|$ s are greater than zero and  $sign(d_1) =$   
 220  $sign(d_2) = \dots = sign(d_P)$ . That is, the granules with continuous motion (in the

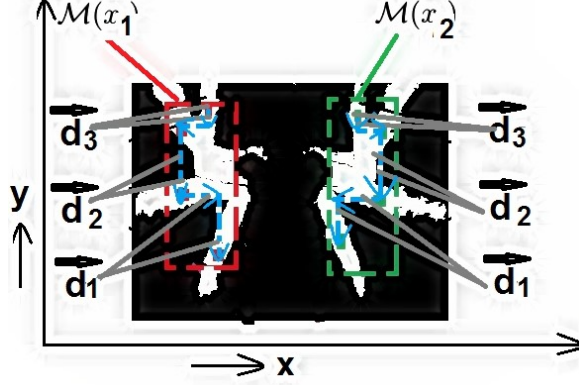


Figure 4: A pictorial example of distance computation following Eqn. 4 over two motion granules  $\mathcal{M}(x_1)$  and  $\mathcal{M}(x_2)$

continuous moving object(s)) will belong to  $\underline{O}_t$  (see Eqn. (5(a))). If there exist at least one pair of points for which  $|d_p| > 0$  in a granule ( $\mathcal{M}(x_i)$ ) and  $sign(d_p)$ s are the same at least twice over  $p \in \{P\}$  then the granule will belong to the upper approximation of the object region in  $f_t$   $\overline{O}_t$  (see Eqn. (5(b))). That is,

225 any object in motion (may or may not be continuous) will belong to the upper approximation of the object set. Similarly, while forming the background set,  $|d_p| = 0$  for all  $p \in \{P\}$  for a granule  $\mathcal{M}(x_i) \in \underline{B}_t$ , where  $\underline{B}_t$  stands for the lower approximation of the background region in the frame  $f_t$  (see Eqn. (5(c))). If there exists at least one point in a granule for which  $|d_p| = 0$  then the granule

230 will belong to the upper approximation of the background region in  $f_t$ , i.e.,  $\overline{B}_t$  (see Eqn. (5(d))). A crisp and definite separation can be performed in this way in the lower approximated regions. However, there exists overlapping in the boundary regions of both the sets. One may note that there is not a common boundary region between object and background sets. Rather, there exist two

235 other regions or two different sets of granules solely characterizing the object boundary and background boundary. That means, there are a few granules which solely belong to the boundary region of the object but do not belong to

the boundary region of the **background**. The reverse is also true for some other granules. Besides, these granules neither belong to the lower approximations of the sets as there exists non-zero probability to belong to its complementary set. Here comes the concept of the 'other class', that is the class which is neither object nor background. The detection of this class can be treated as noise, i.e., no relevant information is present there in these regions even though some motion may be there. This phenomenon of the newly defined object-background set helps to understand of the video more clearly.

$$\underline{O}_t = \{\mathcal{M}(x_i) \in U : |d_p| > 0 \quad \forall p \in \{P\} \quad \& \quad \text{sign}(d_1) = \text{sign}(d_2) = \dots = \text{sign}(d_P)\} \quad (5a)$$

$$\overline{O}_t = \{\mathcal{M}(x_i) \in U : |d_p| > 0 \quad \exists p \in \{P\} \quad \& \quad \text{sign}(d_{pm}) = \text{sign}(d_{pn}) \quad \text{while} \quad \{pm, pn\} \in \{P\}\} \quad (5b)$$

$$\underline{B}_t = \{\mathcal{M}(x_i) \in U : |d_p| = 0 \quad \forall p \in \{P\}\} \quad (5c)$$

$$\overline{B}_t = \{\mathcal{M}(x_i) \in U : |d_p| = 0 \quad \exists p \in \{P\}\}. \quad (5d)$$

A pictorial representation of the rough set (defined in Equation (5)) is shown in Figure 5 in a two dimensional feature space. Here a two class based rough set is shown with overlapping granules. The '+'-class represents the object class, whereas the '-'-class represents the background class. **It is visualized here that how can the object boundary and background boundary regions be not the same, but can have overlapping among each other. Algorithm 1 describes the steps for formation of rough set over video.**

### 3.3. Motion Entropy

This entropy **measures** the amount of certainty in the movements of the object(s). Here we assumed that if there is some probability of an event to take place, the motion pattern of moving/ or static object would face sudden change and thereby causes an uncertainty. **Motion entropy concerns computation of that uncertainty.** It consists of two uncertainty measures, namely, velocity entropy and acceleration entropy. As shown in Algorithm 2, the decision-making

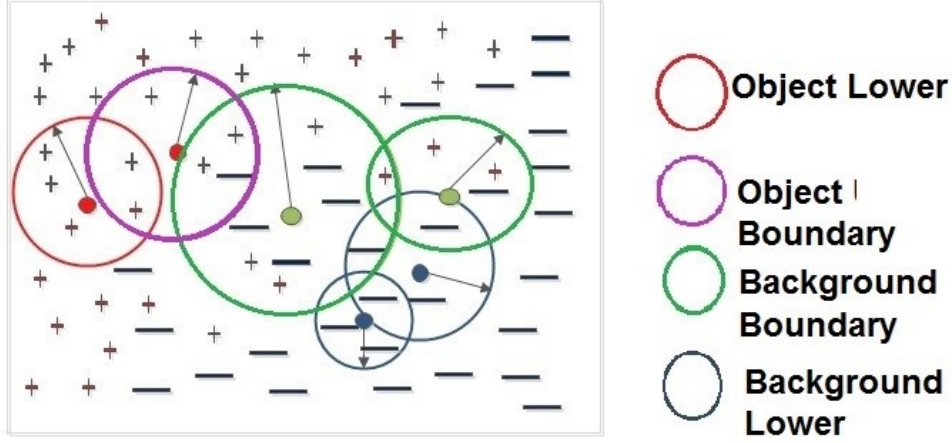


Figure 5: Pictorial representation of a two class object-background Rough Set

regarding the movements is performed based on the nature of shifts from frame  
to frame. Both the static object(s) and object(s) with continuous movements are  
supposed to have low motion entropy values. Whereas the regions with sudden  
change (static-to-moving or moving-to-static) would have higher entropy values.  
The information regarding those regions with uncertainty in the system should  
be updated after observing their nature in the next  $P$  frames. One may note  
that, this entropy is mainly computed over the uncertain regions where motion  
is or may be present, i.e., the regions present in  $\underline{O}_t \cup \{\overline{O}_t - \underline{O}_t\} \cup \{\overline{B}_t - \underline{B}_t\}$ .  
 $Shift$  and  $SC$  (Eqn. (6)) are considered during the formulation of this measure.

Let  $M$  represent a moving object in the  $f_t$ , i.e.,  $M_t \in \overline{O}_t$  or  $M_t \in \overline{B}_t - \underline{B}_t$ .  
Its nature of movement (amount of shifts and changes in shifts) over  $P$  frames are  
stored in two matrices, namely,  $Shift(M)$  and  $SC(M)$ . The matrix  $Shift(M)$   
contains the location change information of moving object  $M$  in consecutive  
frames. The matrix  $SC(M)$  stores the change in  $Shift(M)$  over time. In other  
words,  $Shift(M)$  contains the velocity information and  $SC(M)$  contains the  
acceleration information of the object  $M$ . These matrices are defined in Eqn.

(6).

$$Shift(M) = \{Shift_p : p = 1, \dots, P\} : Shift_p = M_{t-p} - M_{t-(p-1)} \quad (6a)$$

$$SC(M) = \{SC_p : p = 1, \dots, P\} : SC_p = \frac{d}{dt}(Shift(M)) \quad (6b)$$

where  $M_{t-p}$  denotes the location of the moving object  $M$  in the frame  $f_{t-p}$ .

270 One may note that,  $Shift_p$  is also a signed *MahattanDistance* between objects in consecutive frames and computed similarly as of  $\vec{d}_p$  in Equ. 4.

Let the mean values of the matrices be represented as  $Shift_m$  and  $SC_m$ . Let  $ShiftL$  and  $SCL$  be two matrices such that,  $ShiftL = \{s : s \in Shift(M) \& s \geq Shift_m\}$  and  $SCL = \{sc : sc \in SC(M) \& sc \geq SC_m\}$ .

275 The velocity roughness ( $V_R$ ) and acceleration roughness ( $A_R$ ) of that region are computed as:

$$V_R = 1 - \frac{|ShiftL|}{|Shift(M)|} \quad (7a)$$

$$A_R = 1 - \frac{|SCL|}{|SC(M)|} \quad (7b)$$

where  $|\cdot|$  represents the cardinality of the set. Motion roughness is thereby defined as:

$$M_R = \frac{V_R + A_R}{2} \quad (8)$$

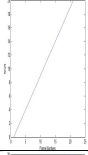

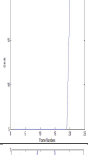

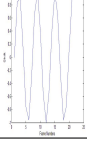
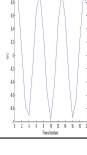
and the motion entropy ( $M_E$ ) is:

$$M_E = M_R * e^{1-M_R}. \quad (9)$$

280 The variations in the values of the uncertainty measures over different types of movements is shown in Table 1 with three examples: continuous movement/ with uniform change in motion (i.e., predictable), random movement, and sudden change in movement.

The moving patterns shown in Table 1 are ideal by nature. In real life this is expected to be more complex. However we can have some ideas regarding 285 the nature of the movements by observing the values of  $M_R$ , and  $M_E$ . We

Table 1: Variations in Entropy with Different Movement Pattern

Movements	<i>Shift</i> Pattern	<i>SC</i> Pattern	$M_R$	$M_E$
Predictable $\in \underline{Q}_t$			0.53	$\simeq 0$
Sudden Change			0.95	0.998
Random $\in \overline{O}_t$			0.5	0.82

can definitely differentiate among the noise and moving object(s) and sudden changes after observing the corresponding  $M_E$ .

One may note that the values of  $M_E$  is expected to be the highest ( $\simeq 1$ ) in case of sudden change. It shows that maximum information will be present there due to its unpredictability. Therefore, while detecting the possible event(s) we set the threshold of  $M_E$  close to 1, i.e., 0.9. On the other hand,  $M_E \simeq 0$  when the movement is predictable, that is very less information is available there. Therefore, we set the threshold of  $M_E$ -value closed to 0, at 0.2 while detecting predictable motion.

#### 4. Conceptualizing the Video by Estimating the Moving Patterns of Objects and Precognized Events

The basic steps lying behind the proposed methodology is shown in Fig. 6. This method mainly consists of two parts. The first one is unsupervised formation of object-background sets and the second one is categorizing the moving segments by observing the nature of movement of the object(s). The formation of rough sets is described in Algorithm 1.



---

**Algorithm 1** Formation of Rough Set over Video

---

INPUT:  $f_t, \dots, f_{t-P}, Thr_t$

OUTPUT:  $\underline{O}_t, \overline{O}_t, \underline{B}_t, \overline{B}_t$

INITIALIZE:  $O_t \Leftarrow \emptyset, B_t \Leftarrow \emptyset$

1: Form first layer granule, i.e., spatio-temporal granules  $\aleph_{sp-Tmp}(x_i)$  according to Eqn. (2).

2: Form second layer granule, i.e., motion granule  $\mathcal{M}(x_i)$  according to Eqn. (3).

3: Compute the the number of points covered by each  $\mathcal{M}(x_i)$  and sort these according to size (large to small).

4: Find the granule  $\mathcal{M}_L(x_i)$  with maximum no. of points  $x_i \in \mathcal{M}_L(x_i)$ .

5: Compute  $\vec{d}_p$  for  $\mathcal{M}_L(x_i)$  according to Eqn. (4).

5: Put it to the any one of the sets  $\underline{O}_t, \overline{O}_t, \underline{B}_t, \overline{B}_t$  according to Eqn. (5).

6: Find the next  $\mathcal{M}$ , ( $\mathcal{M}_{L-1}$ ). Set L-1=L.

**if**  $\mathcal{M}_{L-1} \subset \mathcal{M}_L$  **then**

Remove  $\mathcal{M}_{L-1}$  form the set and go to step 5.

**else**

Set L-1=L and go to step 4.

**end if**

7: Do it for all the  $\mathcal{M}$ .

8: Detect the no. of separate moving objects by computing the spatial nearness among  $\mathcal{M}(x)$ s in  $\underline{O}_t, \overline{O}_t$  and  $\{\overline{B}_t - \underline{B}_t\}$ .

---

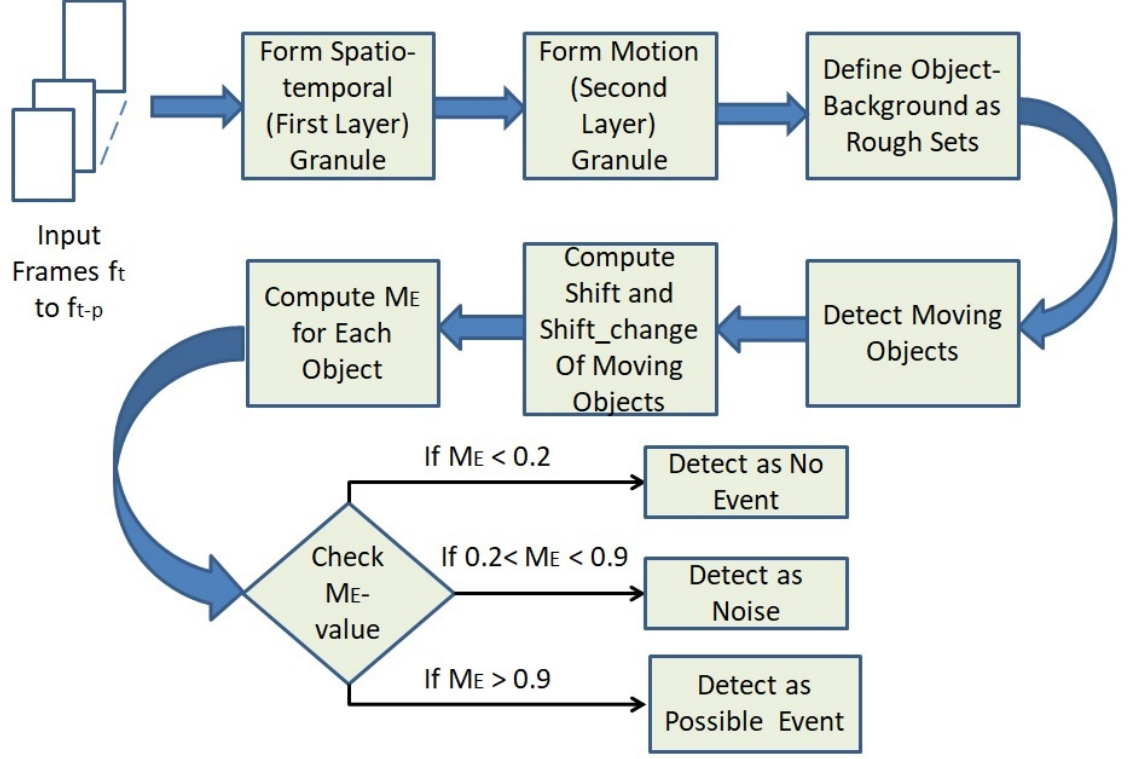


Figure 6: Step-wise Flow Diagram of Video Conceptualization

After the formation of the **object-background** sets the objective is to estimate the nature of motion and change in that nature with respect to time in each set. This is done by mapping and analyzing the frame to frame shifts of each object(s) belonging to the sets  $\underline{O}_t$ ,  $\{\overline{O}_t - \underline{O}_t\}$  and  $\{\overline{B}_t - \underline{B}_t\}$ . No operation will be conducted for the lower background regions, until and unless there occurs some frame to frame deviation in that region. The pattern of movement of certain object is primarily estimated from the initial  $P$  frames and then the regions  $\underline{O}_t$  and  $\overline{O}_t$  get updated alongwith the stream.

Let the  $M^{th}$  moving region in  $f_t$  be denoted by  $M_t$  ( $M_t \in \overline{O}_t \cup \{\overline{B}_t - \underline{B}_t\}$ ). Let the decision attributes be  $D = \{ContMov, RndMov, SuddChng\}$

representing continuous moving (*ContMov*), random moving (*RndMov*), and suddenly changed moving (*SuddChng*) objects. In this work it is assumed that if any sudden change of motion take place there in an object, there is a high probability of some event to occur. The movement pattern estimation and event precognition is done according to the Algorithm 2.

---

**Algorithm 2** Moving Pattern Recognition of  $m^{th}$  Object Till  $t^{th}$  Frame and Event Precognition

---

INPUT:  $M_t, \dots, M_{t-P}$

OUTPUT: Decision regarding the movement pattern

INITIALIZE:  $D \Leftarrow \emptyset$

1: Compute spatial shifts of *Moving* regions.

$$Shift_p = M_{t-p} - M_{t-(p-1)}$$

to form the set  $Shift = \{Shift_k : k = 1, \dots, P\}$ .

2: Compute the change in *Shift* values by computing its derivative over time:

$$SC_p = \frac{d}{dt}(Shift)$$

and where the set  $SC = \{SC_k : k = 1, \dots, P\}$ .

3: Compute *ME* for  $M_t$

**if**  $M_t \in \underline{O}_t$  or if  $M_E \leq 0.2$  **then**

$D = ContMov$

**else if**  $M_t \in \overline{O}_t$  and  $0.2 < M_E \leq 0.9$  **then**

$D = RndMov$

**else if**  $M_t \in \overline{B}_t$  and  $0.9 < M_E \leq 1$  **then**

$D = SuddChng$

**end if**

4: Label  $M_t$  to cause a probable event if  $D = SuddChng$

---

In Algorithm 2,  $Shift_p$ s are signed integers along  $X$  and  $Y$  axes, and therefore it is also a vector like  $d_p$  (in Eqn. (4)) with two elements. The value

represents the shift and the sign represents the direction. Therefore, if the sign  
of the values in the set *Shift* remains almost constant then the deviation in the  
320 set *SC* is very low which reflects continuous motion of the object. Whereas,  
if there is some probability of an event to take place, then a sudden change  
in object motion in  $f_t$  will be observed. It will cause both of the sets *Shift*  
and *SC* have only one non-zero value and those are of same magnitude. The  
325 performance of this algorithm is demonstrated with experimental results in the  
next section.

## 5. Results and Discussions

### 5.1. Preliminary Assumptions and Parameter Selection

The experimental studies on video conceptualization is carried out here are  
330 under the assumption that all the videos are captured by a static camera and  
there is no occlusion or overlapping. Selection of parameters is not much crucial  
issue here. One of the objectives of the study is to make the parameters adaptive,  
as much as possible, so that these can take the approximate values automatically  
depending on the nature of the movement and size of the objects in a sequence.  
335 For example, the value of  $P$  (the number of previous frames to be considered)  
is chosen depending upon the speed of the object. Let  $\tau_p$  be the binarized  
 $T\_V_p$  (see Eqn. (1)) over a threshold  $Th$ . Then, the lowest value for which  
 $\bigcap \tau_p : p = 1, \dots, P \equiv \bigcap \tau_p : p = 1, \dots, P + 1$  is chosen as  $P$ . The threshold value  
 $Thr_t$  used in Eqn (2) is chosen as  $0.3 \times Median(T\_V_p)$ , where *Median* stands  
340 for the statistical median.

### 5.2. Experimental Results in PC

The proposed method is primarily implemented in MATLAB on a PC with  
3.4-GHz CPU. The effectiveness of the proposed method is established in  
this section with experimental results. The main advantages of this algorithm  
345 lies in:

- i) Unsupervised conceptualize of videos by analyzing their nature of the move-  
ments and

ii) Detecting the sudden changes or regions with more information with proposed motion entropy.

350 The video sequences with different characteristics, e.g., indoor/ outdoor surveillance [20, 21, 22, 23], single/ multiple moving object(s) [24, 20, 23], body part(s) movements [25] are considered during the experimentation. All of the video images used here are in RGB color space, however D-feature sensed by kinect sensor [26] is also used wherever it is available ([25, 20, 24]). The algo-  
355 rithm is executed almost over 1000 frames in total, however only a few results are shown to limit the size of the paper.

The video sequences over which the results are shown here have the following characteristics. Sequence  $P - 01$  [21] initially contains one moving person and then different moving cars from different directions appear one by one with  
360 different velocity and with variation in shape and size over the sequence, one of them get stopped and became the part of the background. There are two moving people moving in different direction, stopped, moving only hands, again start moving in  $5b$ -sequence [20]. There is initially one person moving with a bag in a railway platform, where the background train disappears slowly and  
365 the person stops and start to move his hands in  $A - 07$  sequence [22]. Sequence  $Child$  [24] contains one child walking continuously, whereas the people around him moves partially or randomly. One lady is moving her hands continuously with a few moves in the rest of the body in  $M_4$  [25]. Two people enter one by one and have change in their directions, then one of them suddenly sits on  
370 the floor and the other one jumps over him in the sequence  $cam - 132$  [23]. The results that are obtained with these data sets are shown in the following sections.

The visual results are represented as:

- i) Continuous movement/ without any probable event: marked in red color,
- 375 ii) Random movement/ noise: marked in purple color,
- iii) Sudden change/ probable event: marked in green color.

Four frames for each sequence are given in Fig. 7 to demonstrate the process of decision making. It can be seen that the same object(s) is labeled in different

classes in different frames depending on their movement varies. For example,  
 380 the moving car in frame 449 (Fig. 7(1-b)) is labeled as sudden change (moving-  
 to-static) in the frame 669 (1-c), and then to background 823 (1-d). Similar  
 process is executed in the other sequences over objects with similar movement  
 characteristics. The random movements/ newly appeared object(s) are present  
 in scenes (1(a), 3(b), 4(b), 4(c), 5(b)-(d)). The moving-to-static changes/ sud-  
 385 den changes are present there in **scenes** (1(c), 2(c), 3(c)-(d), 4(b), 5(c), 6(c),  
 and 6(d)).

Different motion entropy values with respect to different object(s) in each  
 of the aforesaid sets of frames, **as** shown in Figure 7 are listed in Table 2. The  
 frames are named as those are indexed in the figure, (i.e., from 1(a) to 6(d)), and  
 390 the  $M_E$  values for different regions of a frame are listed in the respective location  
 of Table 2. **It is seen** that the  $M_E$  values for the continuously moving object(s)  
 are very low (marked in 'red' color) is almost  $< 0.1$  for all the cases. Whereas  
 the random movements/ newly appeared object(s) have  $M_E$  values (marked in  
 purple) between 0.6 to 0.85. The  $M_E$  values for the sudden change object(s)/  
 395 or with probable events (marked in green) are always above 0.9. Therefore, our  
 theoretical assumptions are validated with experimental outcomes in this way.

Table 2:  $M_E$  Values for the Frames of Fig. 7

	(a)	(b)	(c)	(d)
1	0.81, 0.04	0.07, 0.02	0.94, 0.06	0.05, 0.02
2	0.02	0.03, 0.74	0.025, 0.96	0.034, 0.026
3	0.033	0.06, 0.81	0.044, 0.069, 0.97	0.075, 0.91
4	0.02, 0.058, 0.043	0.76, 0.07, 0.94	0.73, 0.67, 0.03	0.65, 0.08, 0.02
5	0.036	0.023, 0.78	0.93, 0.78, 0.99	0.82, 0.08
6	0.053	0.033, 0.12	0.97, 0.077	0.98

Identifying the sudden changes present in a video sequence has always been  
 crucial and poses a big challenge. In Table 3 we have shown the accuracy of the  
 said task with our proposed algorithm (Algorithm 2). We have manually identi-

400 fied the frames with sudden change (FSD) for the aforementioned sequences and  
 validated those with the experimental outcomes. The second column of Table  
 3 shows the number of frames that are truly FSD and identified as FSD by our  
 algorithm, i.e., true positive (TP) frames in other words. The third column of  
 this table shows the number of frames that are not FSD but identified as FSD  
 405 by the algorithm, i.e., false positive (FP). The forth column shows the accuracy  
 of the algorithm in identifying sudden change, i.e.,  $TP/(TP + FP)$ .

Table 3: Accuracy of the Algorithm in Identifying Frames with Sudden Change

Name of the Sequence	TP Frames	FP Frames	Accuracy
$P - 01$	65	6	91%
$5 - b$	22	4	84%
$A - 07$	56	5	92%
$Child$	11	2	85%
$M\_4$	65	12	84%
$cam - 132$	78	8	90%

One may note that, the proposed algorithm does not generate any false  
 negative frames. That **means**, the frames with true FSD will always be detected  
 by our algorithm.

410 It is claimed that proposed 'conceptualization' technique will cause compu-  
 tational gain as less number of 'highlight' frames will be required for further  
 processing. We have shown the proof quantitatively in Table 4. It is previously  
 discussed that the frames with sudden change will have the highest entropy  
 value and **they carry** the useful information. Accordingly, we have shown how  
 415 many frames of the aforementioned sequences have motion entropy values  $> 0.9$   
 in the total sequence. From Table 4 we can see that the useful information is  
 present only in 15 – 25% frames of various types of sequences under consid-  
 eration. Therefore, further processing of 85 – 75% frames are not required to  
 interpret the storyline of the videos. This is how we can achieve huge compu-  
 420 tational gain in video understanding.

Table 4: % of Frame Number Reduced From further Processing

Name of the Sequence	Total number of frames	Total number of frames with $M_E > 0.9$	% of gain
$P - 01$	350	65	81.15%
$5 - b$	130	22	85.33%
$A - 07$	250	56	77.6%
$Child$	85	11	87.65%
$M\_4$	65	12	81.5%
$cam - 132$	460	78	83%

### 5.3. Comparative Results

The unsupervised video conceptualization, comprising object tracking followed by event precognition, is a new concept. There is almost no other similar studies to compare its performance. However, tracking based on object-background separation is an underlying part of conceptualization. Therefore we fail to show any comparative studies with the proposed conceptualization method. However, moving object-background separation, described in Algorithm 1 is a part of this work. Therefore we have performed comparative studies with a few recent robust tracking methods over the aforesaid datasets. With this comparative study we will be able to judge how efficient our proposed method is for the task of event precognition, since this tracking performance is crucial behind this task. The tracking methods with which the comparative studies are performed include our earlier method NRBFG [19] and the following six popular methods.

DeepTrack [27]: In this method a single convolutional neural network (CNN) was used for tracking. This network was developed for learning effective features for the target object representation in a purely online manner.

CNT [28]: Here simple two-layer convolutional networks were used for object representations. A 'deep network' structure was then introduced in visual tracking.



KPSR [29]: Key patch sparse representation based tracker was introduced here. Sparse representation, selection of the key patches and designing the contribution factor of the patches were the basic steps of tracking here.

CST [30]: A constrained graph labeling algorithm was proposed here for tracking. Superpixel based transductive learning, the appearance fitness constraint, and the temporal smoothness constraint are incorporated in the graph labeling algorithm which is then used for tracking.

DNNT [31]: A dual deep network is designed here for tracking. Its objective is to exploit the hierarchical features in different layers of a deep model and design a dual structure to obtain feature from various streams. This model is updated online based on the observation tracked object in consecutive frames.

SIMT [32]: This method obtains several samples of the states of the target and the trackers during the sampling process using Markov Chain Monte Carlo (MCMC) method. Here the trackers interactively communicate and exchange information with others thereby improving its performance and increasing the overall tracking performance.

The time and accuracy comparisons among the aforementioned methods to our proposed method is shown in Table 5 . The accuracy is measured based on the distance between the centroids (CD) of the ground truth and the obtained tracked region of the respective frames. The ground truths are available along with the data sets that we used in the experiment. average CPU time in second needed to process a frame.

Table 5 shows that the proposed method results in computational gain as less computation time is required compared to the other methods. It can also be noticed that the accuracy of this method is not always the best among the others. Therefore, the proposed method could be useful where no prior knowledge is available and faster decision making is required. One may note that no visual comparative result is provided here as the proposed method does not show any visual gain compared to the other methods.

Table 5: Time and Accuracy Comparisons

Sequence	Metric	NRBFG	DeepTrack	CNT	KPSR	CST	DNNT	SIMT	Proposed
$P - 01$	CD	4.78	3.22	3.56	3.43	3.82	3.81	3.93	4.95
$P - 01$	Time	0.325	0.305	0.320	0.332	0.381	0.307	0.368	0.165
$A - 07$	CD	4.54	5.94	5.86	5.21	4.68	4.76	4.81	5.21
$A - 07$	Time	0.234	0.323	0.366	0.383	0.265	0.271	0.231	0.189
$Child$	CD	4.23	5.32	4.22	3.57	3.82	3.72	3.61	4.51
$Child$	Time	0.308	0.351	0.385	0.335	0.381	0.344	0.276	0.198
$M\_4$	CD	2.72	2.02	2.42	1.65	1.42	1.55	1.36	2.92
$M\_4$	Time	0.256	0.241	0.295	0.222	0.251	0.246	0.271	0.111
$cam - 132$	CD	4.82	6.02	5.42	4.65	4.72	4.58	4.75	5.22
$cam - 132$	Time	0.276	0.421	0.415	0.322	0.351	0.316	0.284	0.192

#### 5.4. Computational Complexity of Algorithms

It is already demonstrated in Table 5 that the computation time required by our proposed tracking algorithm (Algorithm 1) is much less. Here we are going to show how the computational complexity of this algorithm get reduced with the newly defined motion granules. If there are total  $N$  number of pixels in a frame, and such  $P$  no. of frames are considered for granulation, then the complexity of granulation is  $O(\frac{NP}{n})$ , with  $n$  degree of similarity in three dimensions. Now the object background separation could be done with a single scan through the representative points of the granules. Say, there are  $G$  such granules. Now, since  $G \ll NP$ , the total complexity will still be  $\approx O(\frac{NP}{n})$  in detecting an object. That is why the computation is not much dependent on the speed or size of the objects, and here it varies between 0.11 – 0.19 second/frame.

Now, let us consider Algorithm 2. Let there be  $M$  no. of moving objects present there in the sequence. Then the computational complexity of Algorithm 2 will simply be  $O(M)$  since all the computational procedure will be conducted over the motion granules only.

### 5.5. Experimental Results in IoT

The effectiveness of the proposed algorithm is demonstrated here in an IoT architecture. The architecture is built by embedding a Raspberry Pi 3B+ motherboard, a Raspberry Pi Cam for real time data acquisition, and a breadboard with LED. The codes to implement these algorithms and techniques are written here in Python programming language by leveraging the predefined functions present in OpenCV library, Scipy. The detail of this architecture is shown in Fig. 2 at Section 3. The videos used here for experimentation are both the offline data, as described in the previous section, and the real-time data. The real-time (on-line) sequences were shot by moving a pen with different speeds and directions in front of the Raspberry Pi Cam. Here in Fig. 8, two example frames for off line and on line videos with unusual change detection are shown. LED glows by detecting unusual movements. In case of on-line data, it glows whenever some change is found in the motion/ movement of the pen.

In Table 6 the performance of our algorithm is demonstrated quantitatively. In the first six rows Table 6 is compared with Table 4 in terms of the reduction of frames. The count in the third column of Table 6 is taken as the total number of times, the glowed LED in each sequence, and it is compared with the total number of frames with  $M_E > 0.9$  of Table 4 for the same sequences. It can be seen from Table 6 that the the number of frames where LED glows IoT is almost same as that obtained in the PC. One to two frames only got lost in each sequence due to IoT set up, but the performance is almost similar. Therefore it can be concluded that the proposed algorithm performs well in IoT architecture. The last two rows of Table 6 show the performance of this IoT architecture with two real-time sequences. The real time sequences, as mentioned before, are shot by moving a pen with different speeds and directions in front of the Raspberry Pi Cam. The LED glows when there is some change in motion of the pen. The same sequences were recorded and tested with  $M_E$  values in PC. The forth column shows these results. From the last two rows of Table 6 the similar conclusion can be drawn for real-time sequences as that of the offline sequences. That is, our algorithm is effective with real-time data acquisition too, in IoT.

Table 6: Comparative Study on Frame Number Reduction

Name of the Sequence	Total number of frames	Total number of frames where LED glows	Total number of frames with $M_E > 0.9$
$P - 01$	350	62	65
$5 - b$	130	20	22
$A - 07$	250	55	56
$Child$	85	10	11
$M\_4$	65	12	12
$cam - 132$	460	76	78
$Real - TimeSq - 1$	90	46	48
$Real - TimeSq - 2$	110	32	33

## 6. Conclusions

The process described in this article involves rough set based moving object background classification and motion uncertainty analysis with newly defined motion entropy. The primary goals of this new application lie in, predicting some event to occur, identifying the frames where some events may take place, and selecting a few sets of consecutive frames as the highlight of the entire sequence. The effectiveness of the theories and methodologies described here is experimentally validated with different types of video sequences. The proposed work is also applied and tested in an IoT framework where it proves to work properly even with real-time data acquisition. To our knowledge, no such unsupervised video conceptualization approach exists in literature. The application of event prediction in IoT is also very scarce. Therefore no suitable comparative survey could be provided for the entire task and we showed comparative results are shown for the tracking part only. The new unsupervised tracking method which is a crucial part of the 'conceptualization' task proves to be faster than several recent robust tracking methods. Besides, the motion uncertainty measure, viz., 'motion entropy', defined here can reflect which object(s) present in a video sequence have unpredictable movements, i.e., more suspicious behavior.

Since the method is primarily developed for the videos captured with static cameras, it may not work if camera movement is present there in the video sequence.

This method could be extended to identification of object-of-events and description generation of possible events. Techniques to handle camera motion could also be integrated with this method to make it more robust. Further, the theories and methodologies developed here may be applied in the other areas of computer vision e.g., video summarization and shot boundary detection. Above all, this investigation could be highly effective in the IoT applications with video processing, like smart city, and smart home.

## 7. Acknowledgement

The authors acknowledge Mr. Bharath Y. P. for his help with coding. S. K. Pal acknowledges the Indian National Science Academy (INSA) Distinguished Professorship.

## References

- [1] L. A. Zadeh, Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, *Fuzzy Sets Syst.* 90 (2) (1997) 111–127.
- [2] S. K. Pal, S. K. Meher, Natural computing: A problem solving paradigm with granular information processing, *Applied Soft Computing* 13 (9) (2013) 3944–3955.
- [3] J. T. Yao, A. V. Vasilakos, W. Pedrycz, Granular computing: Perspective and challenges, *IEEE Trans. on Cyberns.* 43 (6) (2013) 1977–1989.
- [4] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Norwell, MA, 1992.
- [5] P. Borges, N. Conci, A. Cavallaro, Video-based human behavior understanding: A survey, *IEEE Trans. on CSVT* 23 (11) (2013) 1993 – 2008.

- [6] M. Brand, Understanding manipulation in video, in: Proceedings of the Second Intl Conf on AFGR, IEEE, Killington, VT, 1996, pp. 94–99.
- 565 [7] C. Stauffer, W. E. L. Grimson, Learning patterns of activity using real-time tracking, IEEE Trans. PAMI 22 (2000) 747–757.
- [8] Y. Yang, J. Liu, M. Shah, Video scene understanding using multi-scale analysis, in: IEEE Intl. Conf. on Comp. Vision, Kyoto, 2009, pp. 1669 – 1676.
- 570 [9] S. Zaidenberg, B. Boulay, F. Bremond, A generic framework for video understanding applied to group behavior recognition, in: IEEE AVSS, Beijing, 2013, pp. 136 – 142.
- [10] T. Milbich, M. Bautista, E. Sutter, B. Ommer, Understanding videos, constructing plots learning a visually grounded storyline model from annotated  
575 videos, in: IEEE ICCV, Venice, Italy, 2017, pp. 4404–4414.
- [11] I. Mademlis, A. Tefas, I. Pitas, A salient dictionary learning framework for activity video summarization via key-frame extraction, Information Sciences, Elsevier 432 (2018) 319 – 331.
- [12] M. Cai, F. Lu, Y. Gao, Desktop action recognition from first-person  
580 point-of-view, IEEE Trans. on Cyberns. DOI 10.1109/TCYB.2018.2806381 (2018) 1–13.
- [13] C. Gao, Y. Chen, J.-G. Yu, N. Sang, Pose-guided spatiotemporal alignment for video-based person re-identification, Information Sciences 527 (2020) 176 – 190.
- 585 [14] Y. Ke, R. Sukthankar, M. Hebert, Event detection in crowded videos, in: 2007 IEEE 11th International Conference on Computer Vision, 2007.
- [15] R. Hinami, T. Mei, S. Satoh, Joint detection and recounting of abnormal events by learning deep generic knowledge, in: The IEEE International Conference on Computer Vision (ICCV), 2017.

- 590 [16] R. Hamaguchi, K. Sakurada, R. Nakamura, Rare event detection using disentangled representation learning, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
- [17] Y.-L. Hsueh, W.-N. Lie, G.-Y. Guo, Human behavior recognition from multiview videos, *Information Sciences* 517 (2020) 275 – 296.
- 595 [18] J. Liang, L. Jiang, J. C. Niebles, A. G. Hauptmann, L. Fei-Fei, Peeking into the future: Predicting future person activities and locations in videos, in: *IEEE Proc. on CVPR*, 2019.
- [19] S. K. Pal, D. Bhunia Chakraborty, Granular flow graph, adaptive rough rule generation and tracking, *IEEE Transaction on Cybernetics* 47 (12) (2017) 4096–4107.
- 600 [20] E. O. W. S. B. J. Davis, V. Sharma, Background-subtraction using contour-based fusion of thermal and visible imagery, *Computer Vision and Image Understanding* 106 (2007) 162–182.
- [21] PETS-2001, *IEEE Int. WS Perfor. Evaluation of Tracking and Surveillance*, 2001.
- 605 [22] AVSS-2007, *Fourth IEEE Int. Conf. Adv. Video & Signal Based Surveillance*, 2007.
- [23] H. Possegger, S. Sternig, T. Mauthner, P. M. Roth, H. Bischof, Robust real-time tracking of multiple objects by volumetric mass densities, in: *IEEE Proc. on CVPR*, 2013.
- 610 [24] S. Song, J. Xiao, Tracking revisited using rgb-d camera: Unified benchmark and baselines, in: *Proceedings of IEEE ICCV*, IEEE, Washington, DC, USA, 2013, pp. 233–240.
- [25] ChaLearn, ChaLearn Gesture Dataset (CGD 2011), California, 2011.

- 615 [26] J. Han, L. Shao, D. Xu, J. Shotton, Enhanced computer vision with microsoft kinect sensor: A review, *IEEE Trans. on Cyberns.* 43 (5) (2013) 1318 – 1334.
- [27] H. Li, Y. Li, F. Porikli, Deeptrack: Learning discriminative feature representations online for robust visual tracking, *IEEE Trans. Image Proc.* 25 (4) (2016) 1834–1848.
- 620 [28] K. Zhang, Q. Liu, Y. Wu, M.-H. Yang, Robust visual tracking via convolutional networks without training, *IEEE Trans. Image Proc.* 25 (4) (2016) 1779–1792.
- [29] Z. He, S. Yi, Y.-M. Cheung, X. You, Y. Y. Tang, Robust object tracking via key patch sparse representation, *IEEE Transaction on Cybernatics* 47 (2) 625 (2017) 354–364.
- [30] L. Wang, H. Lu, M.-H. Yang, Constrained superpixel tracking, *IEEE Transaction on Cybernatics* 48 (3) (2018) 1030–1041.
- [31] Z. Chi, H. Li, H. Lu, M.-H. Yang, Dual deep network for visual tracking, 630 *IEEE Tran. on Image Proc.* 26 (4) (2017) 2005 – 2015.
- [32] J. Kwon, K. m. Lee, Tracking by sampling and integrating multiple trackers, *IEEE Tran. on PAMI* 36 (7) (2014) 1428 – 1441.



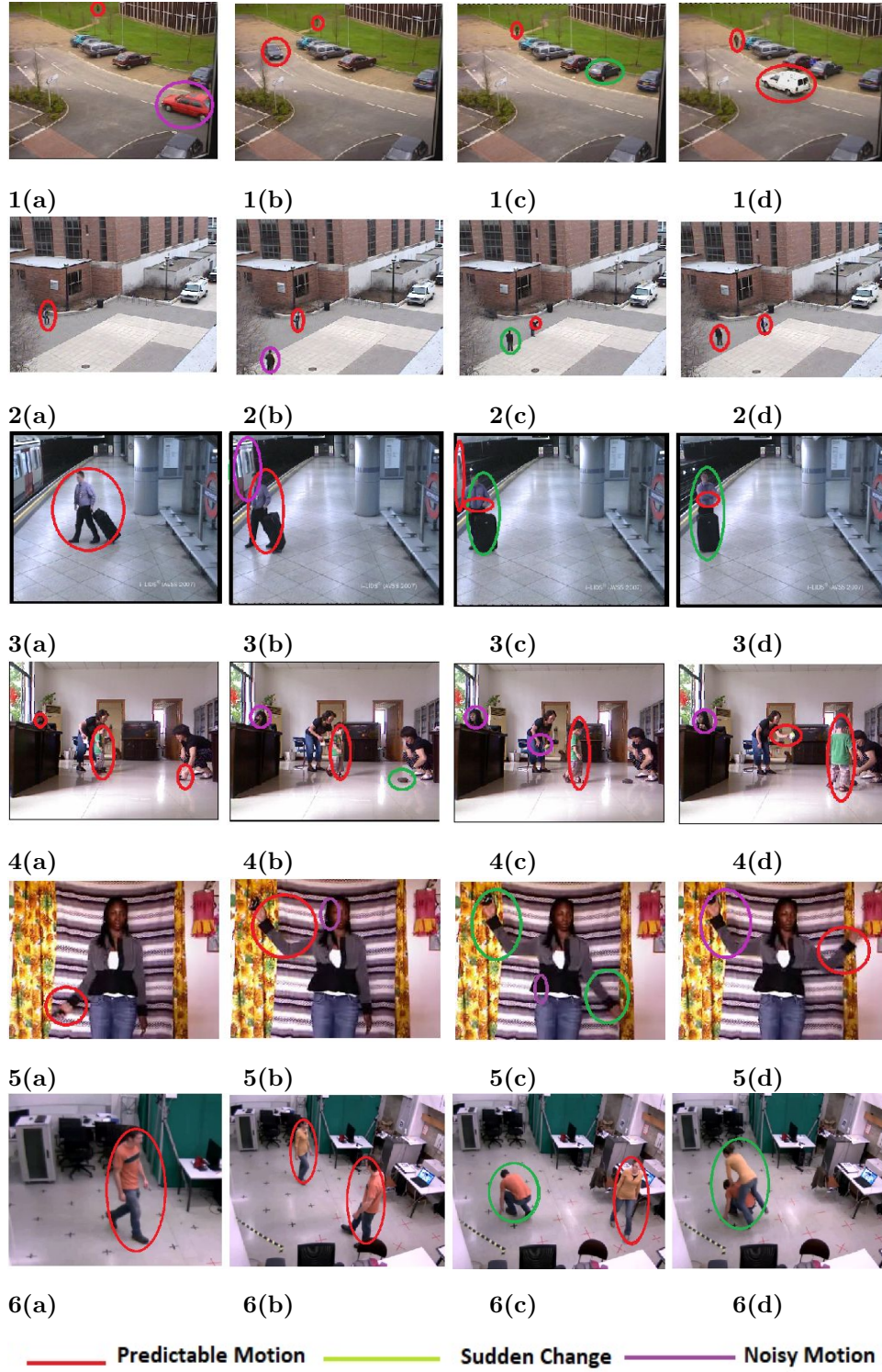
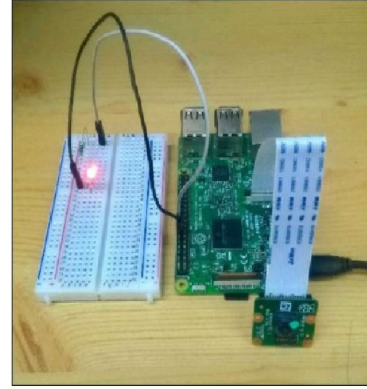


Figure 7: Conceptualization results for frame nos. (1) 12, 449, 669, 823 from *P-01* sequence, (2) 196, 364, 746, 1418 from *5b* sequence, (3) 2339, 2389, 2445, 2492 from *A-07* sequence, (4) 5, 25, 44, 69 from *Child*-sequence, and (5) 7, 16, 21, 28 from *M-4* sequence (6) 159, 239, 310, 365 from *cam-132* sequence



Detection of Unusual Change in Video



LED Glows

(a)



LED Glows by Detecting Unusual Change in Real-time Video

(b)

Figure 8: LED Glows in IoT by Detection of Unusual Change in Motion: (a)Frame no. 692 of  $P - 01$ -Sequence and (b) A Frame in Real-time Sequence