International Journal of Innovative Computing, Information and Control
Volume **17**, Number **6**, December **2021**

 ISSN 1349-4198
pp. 1843–1870

PARTICLE SWARM OPTIMIZATION WITH DEEP LEARNING FOR HUMAN ACTION RECOGNITION

Usman Ahmad Usmani[1,*], Junzo Watada[2], Jafreezal Jaafar[1]
Izzatdin Abdul Aziz[1] and Arunava Roy[3]

[1]Department of Computer and Information Science
Faculty of Science and IT
Universiti Teknologi Petronas (UTP)
Seri Iskandar, Perak 32610, Malaysia
[*]Corresponding author: usman_19001067@utp.edu.my
{ jafreez; izzatdin }@utp.edu.my

[2]Graduate School of Information, Production and Systems
Waseda University
1-104 Totsukamachi, Shinjuku-ku, Tokyo 169-8050, Japan
junzow@osb.att.ne.jp

[3]School of Information Technology
Monash University
Bandar Sunway, Subang Jaya, Selangor 47500, Malaysia
Arunava.Roy@monash.edu

Received May 2021; revised September 2021

Abstract. *Motion target detection and tracking are among the most important computer vision areas that use advanced image processing techniques. To accomplish the purpose of tracking a target, traditional approaches first perform the detection process and then track the target. However, such methods necessitate a constant false-alarm rate system that detects the whole frame obtained at the moment. This decreases the detection efficiency and degrades the target tracking output. Also, the current motion target detection algorithms extract features from the relevant object only if the moving object has complex texture features. The regions extracted by these algorithms are larger than the region of interest and stretches towards the direction of movement. These algorithms are sensitive to noise, and thus, it is difficult to accurately predict the location of the objects. This paper proposes a deep learning framework for human action recognition to overcome the drawbacks of the current state-of-the-art methods. To extract the appearance based and structural information, each frame of the action sequences is evaluated for spatial features. The temporal properties of the video sequences undergo computation across full corresponding blocks frames to give motion based information. The features are reduced using the particle swarm optimization detection technique in video image sequences to reduce the computational complexity. If the scene is stationary, the identification of the moving people is addressed based on the correlation tracking technique. Finally, a deep learning neural network is used to evaluate the method's effectiveness. Since two autoencoders have been trained separately, the information in the autoencoders is forwarded to the deep learning neural network to recognize human actions. Our deep learning method also performs the atomic morphological operations and shadow removal based on the hue saturation value color space. Our method can also track targets that do not have high contrast and prominent features with the background. We infer that our approach helps improve the tracking stability and increases the robustness of the tracking process by an*

DOI: 10.24507/ijicic.17.06.1843

accuracy of 97.09% on the Stony Brook University interaction dataset, 98.02% on the High-level Human Interaction Recognition Challenge interaction dataset, 96.17% on the Weizmann dataset. The statistical measures such as precision (96.76%), sensitivity (95.39%), Matthews correlation coefficient (93.83%), and Jaccard Index (92.63%) are also high, thus demonstrating the better performance than all the current state-of-the-art methods.

Keywords: Particle swarm optimization, Human action recognition, Autoencoders, Deep learning, Human tracking

1. **Introduction.** The key content of the computer vision research focuses on how to use a range of imaging systems to replace human vision as a means of signal input and replace the work of the human brain with computer vision knowledge. The final goal of the research is to make computers observe and understand the environment as humans do through vision. In our previous researches, we have used deep learning in the domain of drone auto navigation [1], enhanced deep learning [2], a comparison [3] to such conventional image processing works as multi-camera tracking [4], particle filtering [5,6,8], clustering [7], meshed method [12], Support Vector Machine (SVM) [10], boosted Histogram of Oriented Gradients (HOG) [9], and panoramic image composition [11]. This work is based on the human action recognition using Particle Swarm filtering Optimization (PSO) technique.

The key goals of human motion vision research are to detect, recognize and monitor the human body from a collection of images including individuals, understand and explain their behaviors. Computer vision technology is used in intelligent video monitoring systems to interpret, understand and process video data recorded by video camera, filter out irrelevant information, useful abstract information, and send it to control employees to deal with it. The system will replace humans in terms of performance by providing early warning, avoidance, and proactive monitoring functions. By human face recognition and gait analysis [13], it is decided if the next person has the right to enter the safe area. Another application focuses on the actions of people on the scene (not only identifying individuality). It is used mainly in the circumstances susceptible to safety criteria such as parking, supermarket, vending machines, Automated Teller Machine (ATM), and traffic management. The device warns the security personnel to prevent crimes when suspicious activity happens at the scene. In addition, human motion analysis in the virtual reality scene has a wide variety of applications such as virtual game, video conference, and figure animation.

The researchers are implementing image search based on sports motion dataset content. They use the vision approach to build the body's geometry model in dance and sports training. They advise and correct the trainer's actions through joint motion analysis to achieve a very intuitive effect. Motion detection, target tracking, and several other methods of motion analysis include the low-level vision module. To solve the merging problem of multi-camera data, we use the intermediate level vision module. The high-level vision module involves identifying the target, comprehension, and specification of the semantic content of motion data. DETER system [15] is used for surveillance of car parks to prevent cars from being thieved. The system can track moving targets, analyze behavior patterns of moving targets according to tracking trajectory, and alarm against abnormal behavior. Human detection, tracking, and behavior judgment are beneficial for pedestrian safety, driving safety, and traffic management. Many functions such as traffic control, car irregular behaviour recognition, pedestrian behavior judgment, and intelligent vehicles have been accomplished [17-19]. Acquiring traffic information will make it more convenient for individuals to use the highway, increase running quality, and relieve traffic congestion. For accident prevention, the identification of suspicious activity

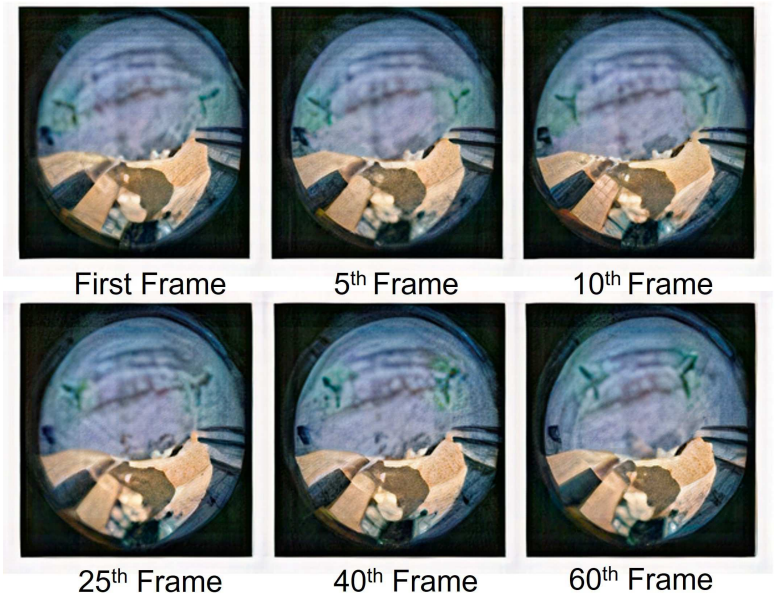


FIGURE 1. Test results using our proposed method. As it can be seen the humans can be seen in a green bounding box and are detected with a high accuracy.

of vehicles is of great importance. It involves segmenting and monitoring the vehicle and then evaluating the tracking of the running trajectory and speed of the car. This helps in determining the occurrence of improper driving. The test results using our proposed method are described in Figure 1. As it can be seen, the humans are detected by a green bounding box.

Motion detection is challenging because of the complex variations of the background image, environmental effect, illumination, shadows, and chaos interference. Currently, the methods in common use background subtraction [14], the temporal difference [16], optical flow [23] calculation and statistical methods [24]. Background subtraction [14,25] methods are most commonly used in motion segmentation. It is a tool that uses the contrast between the current image and the background image to detect a region of motion.

For example, Haritaoglu et al. [14] utilized minimum, maximum intensity values, and maximum temporal difference values to implement statistical modeling for each pixel in scenes and update the background periodically. Lipton et al. [30] used the two frames difference method to detect the moving target from real video images to classify and track the target. Jung and Sukhatme [31] used the continuous frames in image series to implement temporal difference, carried out a real human body motion detection system in moving robot platform, and implemented the detection of the moving body location through the self adaptive particle filter. Another improved method is to utilize three frames difference to replace the two frames difference.

The temporal difference motion detection method is highly sensitive to complex conditions, but sometimes all associated characteristic pixel points cannot be fully abstracted. Therefore, we need techniques to compensate for this problem, such as combining with other image characteristics, and abstracting the human body moving area. Without specialized hardware equipment, it cannot be used to process the full frame video streams in real time. In terms of the human body monitoring system, the next significant problem is identifying the human body target from all moving targets after the moving area data is collected. Various moving regions may lead to different moving targets. For the ease of

human body identification and further study of actions, it is essential to properly identify the moving target. The statistical approach requires a large number of computational and transformation processes that are difficult for the current hardware equipment to process. When the target is not blocked, the value of this approach is high tracking accuracy and stability. The first downside is that since the search area is wider, it needs a considerable calculation. The second is that the algorithm demands that the target deformation is not high and that it is impossible to block the target. Therefore, as templates shift, the region based tracking approach must solve the tracking problem. If moving aim, the change is due to posture change; if target posture change can be processed well, stable monitoring will be done. The difficulty of this method is that how to determine the unique feature set of a moving target. For a limited number of targets, the focus is on retrieving trajectories and models with high precision. The dependency on detailed geometric object models is the most serious drawback of these approaches. To overcome all the drawbacks of the current method, we propose our method for human action recognition by optimizing the features with particle swarm optimization and then testing its effectiveness on our proposed DL based model. In the next part we discuss the significance and the contributions of our method.

Significance of our method for human action tracking. The system's computational complexity increases if the feature space is large. It comprises redundant and irrelevant features obtained in the extraction of features. As a consequence, the number of features is decreased to optimize the system. The evolutionary computation technique [53] has recently been used to select features due to its global searching capabilities. We use PSO selection method and use it to address the problem of local optima. Our method evaluates the features quality and chooses the best features for classification using a multi-objective fitness function. The actions are then classified using a Deep Learning Neural Network (DLNN). Reusing the base model knowledge is a good step for building an accurate recognition system according to the transfer learning principle. This increases our model performance and uses fewer resources, thus helping to overcome the insufficient training data problem. To successfully compress the input, two AE's undergo training independently. The information recorded in the AE's hidden layer is subsequently sent to the DLNN. The DLNN gets the hidden information from the AE's, which like conventional machine learning based techniques, only attempt to learn from scratch. In this research, cross-person identification is also used.

We adopt the temporal difference method, which utilizes the distinction between many neighboring images before and after to abstract the image area of motion. Wavelets are particularly good at representing transients and non-stationary signals. In this case, the human action sequence is seen as a series of time signals, each of which corresponds to a certain spatial location. The approximation coefficients are high for spatial locations associated with 'no change' whereas the detail coefficients are low. However, significant spatial variations lead to a transient signal during an action, resulting in large magnitude detail coefficients. The wavelet coefficient calculation method makes use of a spatiotemporal volume composed of 'T' frames representing the foreground of interacting humans. Before generating the histogram for each frame, the interaction size area is split and normalized into 'N' non-overlapping blocks. In continuous image series, our approach adopts pixel difference between the two adjacent frames. This is best suited for the situation when the background is not a fixed scene. It abstracts the moving area of the human body from the image through the threshold, and then makes the detection outcome more accurate by using the control threshold of the variety model. Then we extract the moving context information based on the statistical pixel characteristics. First, it calculates the

statistical background pixel information (such as color, grayscale, and border), creates a more sophisticated background model using individual pixel or pixel category features, and then dynamically updates the background statistical values. Finally, it divides the image pixel into foreground or background by comparing the current background mode statistical values.

If the scene is stationary, the identification of the moving people is addressed based on the correlation tracking technique. The grayscale images are processed using the texture and characteristic correlation, and the color images are processed using the color correlation. The Number of Squared Differences is the most widely used correlation law. In combination with many prediction algorithms, such as linear prediction, quadratic curve prediction, Kalman prediction, this approach is used to estimate the target position in each frame image and enhance the precision of tracking. Canny [36] operator is used in feature abstraction to acquire edge feature of the target, and SUSAN [37] operator is used to acquire angle and point information of the target. Currently, the commonly used model is tracking based on active contour models or snakes in vision tracking proposed by Kass et al. [38] in 1988. The basic idea is to represent the bounding contour of the object and keep dynamically updating it. Since snake [39] model is just suited to tracking a single target, an active contour model based on level set is used to track multi-target [40-42]. Our human tracking based model describes humans with different visual angles and postures by using a set of time varying posture parameters.

Thus summing up, we track human behavior by developing a safety monitoring system based on DL around a heavy duty truck. We identify the moving target from the video image series, recognize the human body, and monitor moving humans. Figure 2 shows a high-level overview of the proposed framework and provides a detailed explanation of the

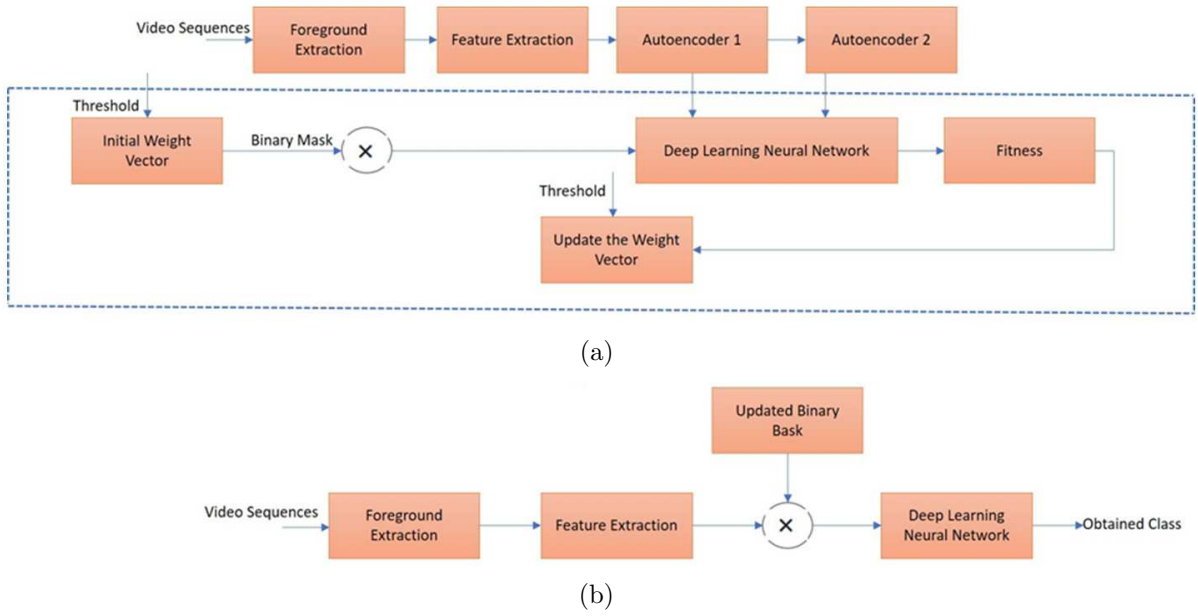


FIGURE 2. Foreground extraction is done for the video sequences. The features are extracted and fed to the two Autoencoders (AE's). The information from the autoencoders is fed to the DL Neural Network (NN) for predicting the effectiveness of our method. (a) shows the training of our DL algorithm and (b) describes the testing.

proposed system for recognizing human actions, wavelet based feature extraction, ROI extraction, and the DLNN (pre-trained) action classification. The significant contributions of the paper are as follows.

1) First, we processed videos taken by a Fisheye camera (FOV) because of its wide field of view. To reduce the computational expense, we use a Restricted Boltzmann Machine (RBM) based AE to help in pre-training the DLNN.

2) Second, on ground video sequences, three separate context modeling approaches are tested: frame difference, approximate median, and Gaussian mixture. The Canny Operator edge detection points and wavelet transform coefficients undergo computation in the spatio-temporal volume to identify activities of interest in a video feed.

3) To monitor the moving human combined with the shadow removal method, we apply the mathematical morphology method and area filling on the detected moving region.

4) We employ a DLNN to evaluate the method's effectiveness. To find the structural or appearance based information, each human sequence action or every frame is evaluated to calculate the spatial features. The temporal properties of sequences of videos undergo entire frames computation of matching blocks to give motion information.

5) We conclude that our approach improves tracking stability and robustness by 97.09% for the Stony Brook University (SBU) interaction dataset, 98.02% for the High-level Human Interaction Recognition Challenge (UT) interaction dataset, and 96.17% for the Weizmann dataset.

The article's structure is as follows. Section 2 presents the current state-of-the-art methods for human tracking. Section 3 gives the proposed method and the technical contributions of our paper. Section 4 provides a detailed overview of our method and provides the results and discussion of our method. Finally, in Section 5, we conclude our article.

2. **Related Work.**

Tai et al. [20] developed a real video tracking system for traffic surveillance and accident detection in crossroads, detecting moving vehicles and automatically analyzing the running trajectory. Pai et al. [21] proposed a method for human detection and tracking in crossroads to ensure safe driving and pedestrian safety. The research aimed to introduce the automotive driving of automobiles. It used the video camera mounted on the vehicle to detect and control the vehicles, pedestrians in front of the road. For example, SAVE-U [22] is a significant innovative European project to develop an integrated safety concept for VRUs such as pedestrians and cyclists. Alzahabi and Cain [25] utilized the self adaptive background model combining pixel color with gradient information to solve the impact of shadow and unreliable color clues for segmentation. Abdelrahim et al. [27] and Chebi et al. [28] adopted a self adaptive background model based on Kalman filtering to adapt to the temporal changes of climate and illumination. Stauffer and Grimson [29] utilized a self adaptive mixture of Gaussian's background model (establish the model for each pixel using a mixture of Gaussian distribution) and an updated model using online estimation to process the impact illumination change, interference of chaotic background motion reliably.

Another improved method is to utilize three frames difference to replace two frames difference. Maddalena and Petrosino [16] developed a mixture algorithm combining self adaptive background subtraction with three frames differences which can detect moving targets effectively and quickly from the background. Motion detection technique based on optical flow measurement utilizes optical flow field features of moving target that shift over time to abstract and effectively track the moving target. The benefit of this approach is that the moving video camera can detect a substantial moving target. The downside is that the method of measurement is very complicated, and the anti-noise

efficiency is low. The visual tracking method is divided into four categories in some literature [34,35] as region-based tracking, feature-based tracking, deformable-template-based tracking, and model based tracking. The basic principle of region-based tracking is to use a pre-determined man-made method or image segmentation method to obtain a prototype containing targets and use similar algorithms to detect targets in a series of images.

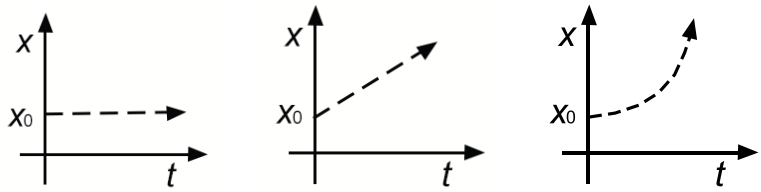
In combination with several prediction algorithms, such as linear prediction, quadratic curve prediction, and Kalman prediction, this approach can estimate the target position in each frame image and enhance the precision of tracking. Generally, there are three kinds of models [23] as line graph model, two-dimensional model, and three-dimensional model. The most widely used is the three dimensional model based tracking system. There are two commonly used target classification methods: shape based classification [30], and it utilizes the shape features of detected moving regions to classify the targets. Another method is motion based classification, and it uses the periodicity of the human body moving to classify the targets [33].

The researchers identified that occlusion is a complicated issue based on their survey [45-47]. The machine should recognize partial and complete object occlusion during any attempt to monitor an object. Some devices can lose data on the object tracking task after occlusion has occurred. It is made worse because it takes a huge system to determine the trajectory of an object as it travels in the field in the image plane. For example, the illumination that happens when light, such as the light of a car, is reflected from different sources, from sunlight to darkness. The detection method is impaired and made more complicated [48,49] as the illumination of the target varies. Object motion is usually present at different degrees of scale [50,51]. Often its movement is swift and abrupt. Using a statistical approach makes it difficult to calculate the velocity of motion required to track the object. Any background noise [52,53], interference, and confusion present in the field must be able to withstand a realistic device. The movement of the object and from other sources creates noise and interference. These variables can lead to mistakes and confusion in identifying the targeted object. It is difficult to detect and track an object and represent the object in a complex shape.

This is due to the existence of the moving object in which its position [54,55] can move freely, rotate and shift. It is important to note that the detection algorithm must separate the foreground and context. This is particularly so when the image is against the backdrop of "messy" [56,57]. In addition to this criterion, it is essential for the detection algorithm to be able to differentiate similar looking objects such as shadows [58,59] from the real object. Device tracking and surveillance deal with detecting the movement of many people [60,61] in crowded spaces, such as a train station, an airport, or a highway. The systems must be able to track several targets for these purposes. In a hectic environment, the tracking system must track and detect any object's movement in a group of moving objects. Applications in virtual reality are immersive and capable of recording high speed motion information that has a high data rate. In the real-life situation [44,62,63], the processing time is an essential device necessary to ensure a good interpretation. Numerous researchers have proposed different methods for object tracking based on the above problems. The approach to be used depends on where they are in the background or setting and what the specifications are for.

Monitoring applications are getting more advanced and more affordable now. Researchers have suggested many methodologies for monitoring applications. The general technique of a tracking device is shown in Figure 3 graphically. It is concluded that the tracking application methodologies are divided into three significant classes based on the vision survey: detection process, tracking process, and object representation process [44].

FIGURE 3. General methodology of a tracking system



(a) (b) (c)

FIGURE 4. Traditional motion models [64]. x_0 and x are defined to be variables and in (a) the line $x = x_0$ is a line parallel to the x axis, (b) the equation is a linear equation where v_0 represents the velocity and t is the time. The value of $x = x_0 + v_0 t$ is the sum of x_0 and the product of v_0 and t. (c) a_0 is the acceleration. The value of $x = x_0 + vt + a_0 t^2/2$ is the final sum of all the three values computed in the equation.

The tracking of an object's movement is a very challenging task since the target motions are dynamically changed. In the conventional method, an object motion can be measured by two models: 1) stochastic models with deterministic and random components, and 2) stochastic model by classical deterministic mechanics [59]. Figure 4 shows an example of such a model that can be expressed using one of Figure 4(a) constant position, Figure 4(b) constant velocity, and tracking an object motion based on constant acceleration model as shown in Figure 4(c) is quite difficult since these two models are not in linear motion. Furthermore, the size of an object will be affected during movement in the camera view. A lot of techniques and methods have been developed to overcome these problems. However, to decide the best technique for object tracking, some criteria of tracking features must be considered.

A significant element in tracking applications is the selection of an object function. Basically, an object feature should be unique and can be differentiated from other objects and feature space, particularly in multiple tracking systems. The object function is usually selected manually based on the tracking application. There are four commonly used object characteristics in the application of tracking: 1) edges, 2) color, 3) optical flow, and texture. The specifics of the visual features listed above are as follows [65,66]. Edges or line features in an image are widely used. Inherently, the most common method to detect significant discontinuities in the grey level is edge detection. In other word, this technique is capable of extracting the outline of the image in various areas, which consists of pixels that have something in common. The key benefit of the edge feature approach in the tracking framework is its lower sensitivity to lighting changes than the approach to color features [67,68]. Color is derived from the distribution of light energy versus wavelength in the real case. A color image serves as distribution information for each image pixel in image processing and can be a descriptor for defining an object and extracting an image attribute.

The RGB (Red, Green, Blue) color space is generally used in computer displays in terms of digital image processing, but other spaces are also used in different contexts, such as CMY (Cyan, Magenta, Yellow), CMYK (Cyan, Magenta, Yellow, and Key) and HSV

(Hue, Saturation, Value). In brief, different numbers of color spaces have been used in the application monitoring, such as [69-71]. However, in using this method, the key problem is that it is too sensitive to illumination [72,73]. Therefore, the researchers have combined the color function with other characteristics [74,75] to overcome this problem. Identifying the texture of the material is one of the most significant approaches to area description. The texture is an asset that depicts an image's surface and structure. Generally, frequent repetition of an element or pattern on a surface can be represented as texture. It can be seen as a grouping of similarities in an image, or its features are continuously and steadily changing or roughly periodic. The texture can be split into two categories: 1) stochastic texture and 2) texture of the structure. It is possible to define the image of stochastic textures as noise, such as color dots randomly scattered all over the image. It is barely characterized by the minimum and maximum brightness and average color attributes. Grass, flowers and so forth are an example of a stochastic texture. On the other side, the texture of the structure is seen as a replication of similar patterns.

The propagation of apparent motion based on velocities or acceleration of the brightness pattern in an image is optical flow [75]. Therefore, it reflects only those changes in the picture in motion related intensity. In this technique, all feature images are expressed in optical flow known to flow detection errors. However, an optical flow is not prone to changes in illumination and motion of unimportant artifacts such as shadows. The different tracking methods for video data can be grouped into three groups in the computer vision literature:

- Region-based tracking
- Active contour-based tracking, and finally
- Feature-based tracking.

The main objective of the region based approach is to discover the region of the object in each frame. An object has a complex shape in the tracking system and is difficult to explain in simple geometry. Therefore, to solve this problem, the region based approach can be used. This is triggered by the ability of this technique to provide these objects with an accurate shape definition. The tracking based on area is used in the tracking algorithm as primitives. The term "regions" refers to the linked elements of points extracted from a stage of motion based segmentation. In relative motion with respect to the camera, the region can be perceived as the silhouette of the projection of an object in the image. A successful contour based tracking is a tracking technique based on an object's boundaries. It is evident from the above statement that this tracking method is executed based on the edges of the intended object. This approach has been established to overcome the computational complexity issue that is usually faced in region based tracking.

Another alternative to the monitoring method is feature based tracking. This approach aims to extract local regions of interest (features) and classify each sequence image's corresponding characteristics. Color, texture, object actions, etc., may be assigned to the attribute of an image. This method enables the researcher to monitor an object according to an object's entire or specific parameters (sub-feature). Generally, all the aforementioned tracking approaches can be carried out by two methods: tracking by detection and statistical tracking.

3. **Proposed Method.**

This section discusses our proposed human action recognition system in detail. The extraction of ROI, wavelet-based feature extraction, and pre-trained DLNN action classification are given in detail in this part.

3.1. **Foreground extraction.**

The proposed method starts with segmenting the foreground image. The foreground image in a static-based environment is created when each

frame is removed from the background model. The Gaussian Mixture Model (GMM) with the expectation-maximization technique separates the foreground if the background is dynamic and complex. Then, morphological techniques such as erosion and dilation are employed to remove any extraneous elements from the foreground image. The following criteria are used to identify the supporting ROI. The ROI is defined as a rectangular type region containing vertical and run sides across the object's centroid if two items in the segmented image are recognized as spatially separated. The interaction region of the objects overlapping is shown in Figure 1 as a rectangle with a width half that of the overlapped zone and a centroid corresponding to the foreground object (c). The green rectangles represent the ROIs in Figure 1. The region is limited to a particular location where arm movement occurs rather than gathering data from the whole contact zone.

3.2. **Extraction of features.** This section deals with the extraction of spatial and temporal characteristics from the visible ROI. To extract the appearance-based and structural information, each frame of the action sequences is evaluated for spatial features. The temporal properties of the video sequences undergo computation across full corresponding blocks frames to give motion-based information.

3.2.1. *Spatial feature extraction.* The Canny edge detector operator is the best step-type edge detector operator, which Canny proposed in 1986 [8,9]. The Canny edge detection technique steps are as follows. The image is smoothed using a Gaussian filter. The Gaussian function is described in Equation (1) as follows:

$$G(x) = 1\Big/\sqrt{2\pi\sigma}e^{-r^2/\sigma^2} \tag{1}$$

where σ denoted the Gaussian curve's standard deviation and reflected the smoothness. We are using the finite-difference of 2×2 neighborhood first order partial derivatives and calculate the gradient direction and gradient magnitude of the smoothed data array $I(x, y)$ calculated in Equation (2) and Equation (3) as follows:

$$|G| = \sqrt{|G_x^2| + \left|G_y^2\right|} \tag{2}$$

$$\theta = \arctan(G_y/G_x) \tag{3}$$

We then use a non-maxima suppression method [6,7] for calculating the gradient magnitude. The following Equation (4) below represents the instances demonstrating the non-maxima suppression:

$$N[i, j] = NMS(M[i, j], \xi[i, j]) \tag{4}$$

We use a double-threshold technique for connection and edge detection. The threshold value is a critical attribute in Canny operator edge extraction [8]. The appearance of a border happens on and off when the threshold is surpassed. If it is set too low, false edges will appear [9]. We utilize a double-threshold approach to solve the threshold setting issue. The adaptive method, based on the gradient histogram and in-class minimal variance [10], is used to calculate both the high and low thresholds. Because the process can determine the threshold based on different images, this method does not need human participation. First, we enhance our technique for calculating the gradient magnitude of an image. The pixel gradient amplitude is determined by computing the finite difference of the first-order partial derivatives of the x, y, 45°, and 135° directions across an area of 8 pixels. The calculation formula is shown in Equation (5) and the function representation is shown in Equations (6)-(10) as follows:

$$M[i, j] = \rho P_x[i, j]^2 + P_y[i, j]^2 + P_{135}[i, j]^2 + P_{45}[i, j]^2 \tag{5}$$

where:

$$P_x[i,j] = I[i+1,j] - I[i-1,j] \tag{6}$$
$$P_y[i,j] = I[i,j+1] - I[i,j-1] \tag{7}$$
$$P_{135}[i,j] = I[i+1,j+1] - I[i-1,j-1] \tag{8}$$
$$P_{45}[i,j] = I[i-1,j+1] - I[i+1,j-1] \tag{9}$$

The gradient direction is given by Equation (10) below:

$$\theta[i,j] = \arctan(P_y[i,j]/P_x[i,j]) \tag{10}$$

To compute the dynamic threshold value, the whole image is segmented into several subimages. Since the sub-image regions may overlap with the scale parameter of the overlap region accounting for the sub-image being ρ. The high and low thresholds of the sub-images are then adaptively adjusted depending on the non-maxima suppression results. We create adaptive dynamic thresholds based on the global edge gradient feature information and the edge gradient information of each sub-image. In images, the proportion of non-edge is often much higher than the percentage of edges. The gradient of all the pixels in the sub-image connected to $\sigma_{\max}$ determines the variance of all the pixels in the sub-image linked to $\sigma_{\max}$. The gradient of all the pixels in the sub-image related to $\sigma_{\max}$ is used to compute the variance of all the pixels connected to $\sigma_{\max}$ in the sub-image.

$$\sigma_{\max} = \sum_{i=0}^{k} (H_i - H_{\max})^2 \Big/ N \tag{11}$$

The total number of pixels in the gradient is N, and the maximum number of nonzero pixels is k. Each sub-high image's h threshold value is chosen outside of the gradient histogram's non-edge area. Otherwise, the final output will have a lot of pseudo-edge noise. The formula for determining τ_h and τ_l is given in Equations (12) and (13) as follows:

$$\varsigma_h = H_{\max} + \sigma_{\max} \tag{12}$$
$$\varsigma_l = 0.4 * \varsigma_h \tag{13}$$

Assume τ_h and τ_l are the total high and low threshold values. The process for obtaining access to them is the same as it was before. Finally, we split the high and low thresholds as follows on the sub-image given in Equations (14) and (15) as follows:

$$\varsigma_{high} = (1-\beta)\varsigma_h + \beta\varsigma_h \tag{14}$$
$$\varsigma_{low} = (1-\beta)\varsigma_l + \beta\varsigma_l \tag{15}$$

where $0 < \beta < 1$, β represents the adjustment rate threshold. If $\beta = 0$, the image segmentation is done using the global gradient histogram's features and does not need to be modified. If $\beta = 1$, it has been fully segmented using the sub-partial image's attributes. Finally, on isolated locations, we conduct boundary tracking and noise reduction.

3.2.2. *Temporal correlation using wavelet transform based feature extraction.* Wavelets are excellent at describing transients and non-stationary signals. In this instance, the human action sequence is seen as a series of time signals, each corresponding to a particular spatial location. The approximation coefficients for spatial locations linked with 'no change' is high, while the detail coefficients are minimal. However, during an action, substantial spatial variations contribute to a transitory signal, resulting in high magnitude detail coefficients. The wavelet coefficient computation technique uses a spatio-temporal volume made up of 'T' frames representing the foreground of interacting humans. Before the histogram is generated for each frame, the interaction size area is divided and normalized

into 'N' blocks that do not overlap. G_p is calculated by organizing the pth block in every frame containing the spatiotemporal-based volume of 'T' total frames taken in a sequence of time. G_p is represented as $[h_{p0}, h_{p1}, \ldots, h_{p(T-1)}]$. The wavelet coefficient computation and G_p are shown in Figure 5. As seen below, G_p's wavelet transform is represented by G_p in Equation (16) as follows:

$$C_p = \left[C_p^{A3}; C_p^{D3}; C_p^{D2}; C_p^{D1}\right] \tag{16}$$

C_p dimensions are $T \times 1$. The notations C_p^{D2}, C_p^{A3}, C_p^{D3}, C_p^{D1} denote the approximation and detail coefficients obtained at levels 1, 2, and 3 for the pth block of video sequences, respectively. The feature vector 'C' of dimension $TN \times 1$ is created by organizing the wavelet coefficients retrieved in the spatiotemporal volume of the 'T' total frames for each block as shown in Equation (17) as follows:

$$\begin{aligned} C = \big[&C_0^{A3}, C_1^{A3}, \ldots, C_N^{A3}, C_0^{D3}, C_1^{D3}, \ldots, C_N^{D3}, C_0^{D2}, C_1^{D2}, \ldots, C_N^{D2}, C_0^{D1}, \\ &C_1^{D1}, \ldots, C_N^{D1}\big] \end{aligned} \tag{17}$$

The vectors C and β after formation undergo concatenation for the formation of feature vector 'X' having dimensions as shown in Equation (18) as follows:

$$1 \times \lambda(T = \lambda(2M + N + L)) \tag{18}$$

3.3. **Optimal feature space selection using Particle Swam Optimization (PSO).** Even though the coefficients wavelets that are orthogonal help distinguish the distinct action sequences, decomposition of wavelets does not decrease dimensions on its own. Consequently, for extracting the optimum wavelet coefficients to reduce the dimensionality, a PSO-based approach is used. The subset of features is chosen from feature-based reduction techniques for performing better in the whole set. The objectives of increasing classification performance (lowering classification error) and decreasing feature count, on the other hand, are incompatible. For solving the issue, the method proposed adopts a multi-objective fitness PSO approach.

In this work, the PSO optimization technique is used to choose features. Given a set of attributes $X = x_1, x_2, \ldots, x_\lambda$, the generation of a set of weights by the PSO as weights $= \theta_1, \theta_2, \ldots, \theta_\lambda$ after several generations. By thresholding PSO's weight vectors, the binary mask required for the feature reduction task is created. The position vector taken for the ith swarm is $\theta_i = (\theta_{1i}, \theta_{2i}, \ldots, \theta_{\lambda i})$, with the number of features in the feature space equal to the number of features in the feature space. $V_i = (v_{1i}, v_{2i}, \ldots, v_{\lambda i})$ describes the related velocity vectors, initialized at random using a uniform distribution. The θ_d vector is used in the generation of a reduced feature space based on Equation (19) listed below.

$$x_{kd} = \begin{cases} x_{kd} & \text{if } \theta_{kd} > \delta \\ 0 & \text{if } \theta_{kd} < \delta \end{cases} \tag{19}$$

The indication of number '1' reflects the feature value taken for the particular location which undergoes selection. The deep learning neural network is fed a reduced set of data for computing the value of fitness. The reduced feature set and classification performance are prioritized using the multi-objective fitness function [10,51]. The symbol represents the fitness function in Equation (20) as follows:

$$\begin{aligned} f = \alpha * &\frac{\textit{No. of features in the selected set}}{\lambda} \\ &+ (1 - \alpha) * \frac{\textit{MSE of selected feature set}}{\textit{MSE of all available features}} \end{aligned} \tag{20}$$

The velocity and location of the particles undergo modification for every 't' iteration depending on their fitness or self fitness of their neighbors, as shown in Equations (21) and (22) as follows:

$$\theta_i^{t+1} = \theta_i^t + v_i^{t+1} \tag{21}$$

$$v_i^{t+1} = \mu * v_i^t + \delta_1 * \sigma_1 * \left(p_i - \theta_i^t\right) + \delta_2 * \sigma_2 * \left(p_g - \theta_g^t\right) \tag{22}$$

where μ is the inertial weight used in controlling the previous impact of velocities relative to the current velocities. These constants reflect the acceleration constants δ_1 and δ_2. σ_1 and σ_2 are two integers generated at random from the range $[0, 1]$. Let p_g and p_i represent the global and local best particles, while p_{gd} and p_{id} represent the global best particle and local best elements. The maximum speed is set in the range $[0, 1]$. The local best has been changed to highlight the importance of both the number of characteristics and the accuracy of classification described in Equation (23) as follows:

$$p_i = \begin{cases} \theta_i & \text{if } f(\theta_i) > f\left(p_i^{prev}\right) \text{ and } |\theta_i| \leq |p_i^{prev}| \\ \theta_i & \text{if } f(\theta_i) = f\left(p_i^{prev}\right) \text{ and } |\theta_i| < |p_i^{prev}| \\ p_i^{prev} & \text{elsewhere} \end{cases} \tag{23}$$

The p_g global particle is considered best and is believed to be the biggest particle present locally and discovered. If the current global particle is better than the current local particle, it is updated iteratively. The velocities and particles are repeatedly changed based on their own and the neighbors' experiences. The achievement of global minimum in the preceding iteration is the feature weight vector that is the most optimal. The binary mask is finally constructed for demonstrating the selection of features at binary value one and rejection everywhere else.

3.4. **Deep learning neural network.** The basic model of a deep learning neural network, an autoencoder, is shown in Figure 5. As demonstrated in [11], the network with two AEs stacked together is utilized in the proposed research. The two AEs are trained

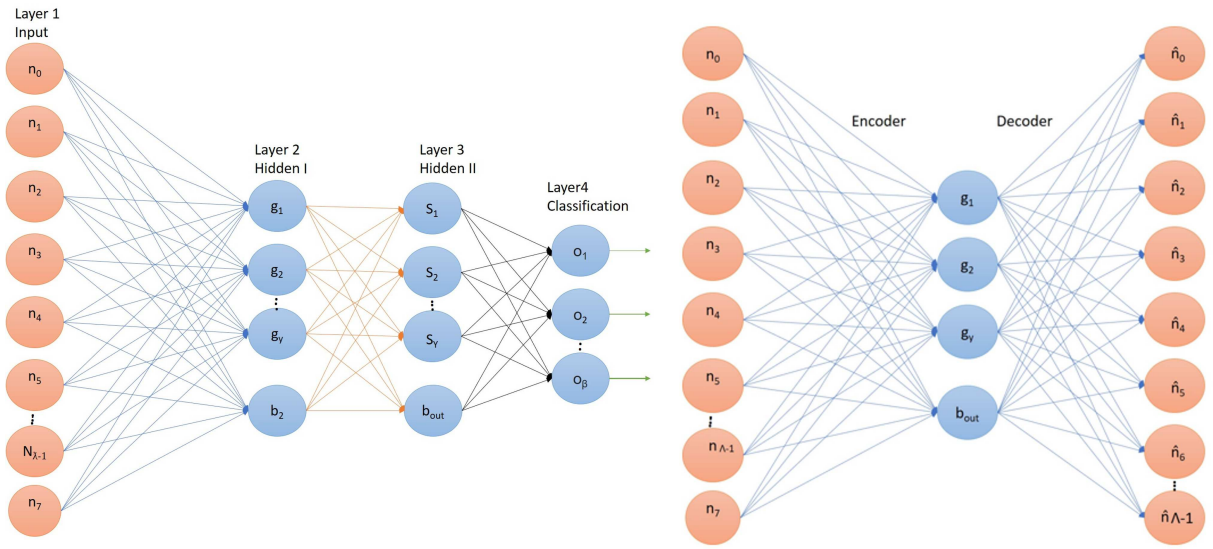


FIGURE 5. The structure of (a) A DLNN having stacked AE's and (b) AE. An AE is a kind of Artificial Neural Network (ANN) that learns efficient codings from unlabeled input (unsupervised learning).

individually and consecutively. The first AE receives the 'X' input feature vector, and it is trained similarly to [11]. The first AE's hidden layer is then utilized for training the second AE. The first and second AE's hidden layers are denoted by g_i, $i = 1, 2, \ldots$ and s_i, $i = 1, 2, \ldots, \Gamma$, respectively, in Figure 5(a). The 'X' feature vector serves as the deep learning network's input, while the classification layer is the output. The deep learning network's hidden layers are made up of the two (trained) AEs. The autoencoder training method is given here. In the autoencoder [42], the vector input $X \in R^{1*\lambda}$ goes encoding in vector Ψ, $\Psi = [r_1, r_2, \ldots, r]^T$, using Equation (24) as follows:

$$\Psi = \Phi(WX + b_{IN}) \tag{24}$$

The input and hidden layers are linked by the weight vector $W \in R^{\gamma \times \lambda}$, the bias vector b_{IN} serves as the bias vector, and the sigmoid function $\Phi(z)$ is specified as $\Phi(z) = 1/(1 + \exp(-z))$. The formula in Equation (25) is used to re-create the AE output $\vec{x}$ and the re-construction is done from Ψ in Equation (25) as follows:

$$\Psi : X = \Phi(Vt\Psi + b_{OUT}) \tag{25}$$

The bias vector is denoted by b_{OUT}, while $V \in R^{\gamma \times \lambda}$ denotes the weight vector connecting the hidden and output layers. By lowering the sparse mean square error allows the AE's weights and bias to be learned in Equation (26) as follows:

$$\xi = ||X - X_b||^2 + \Lambda_1 \Omega_{weights} + \Lambda_2 \Omega_{sparsity} \tag{26}$$

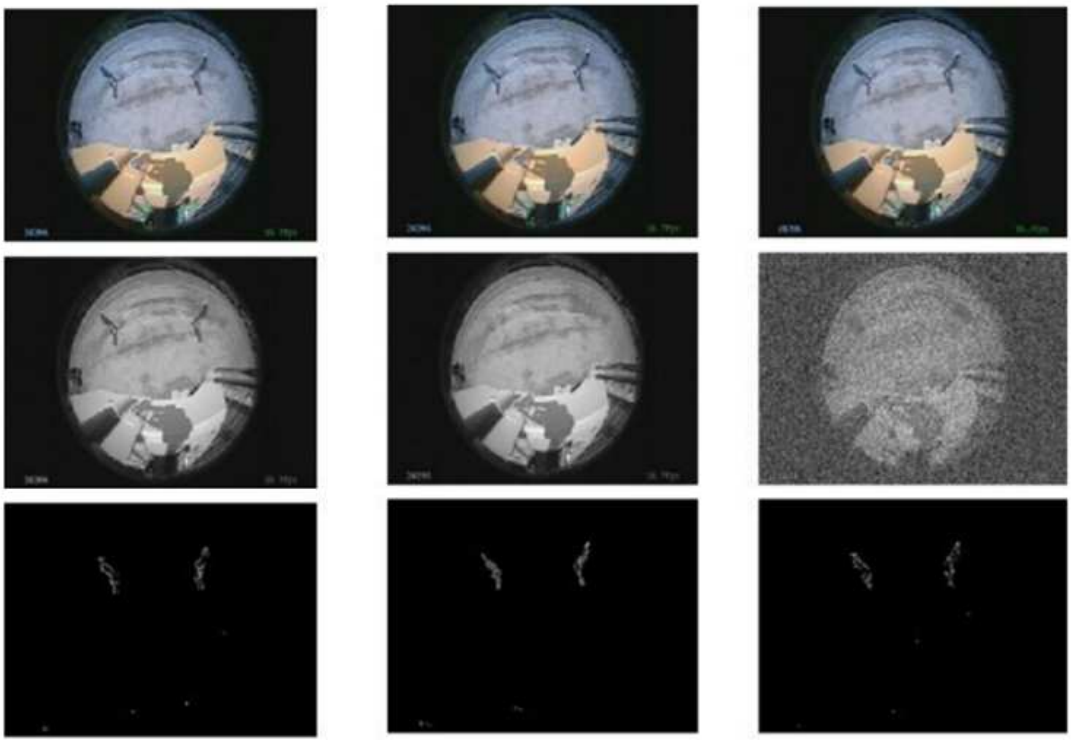
The variables $\Omega_{sparsity}$ and $\Omega_{weights}$ in the above equation correspond to sparsity regularization and L2 regularization, respectively. Λ_1 and Λ_2, respectively, represent L2 regularization and sparsity regularization coefficients. A higher sparsity degree is produced by low values of the sparsity percentage Λ_2. The regularization effect is improved by generating negligible weight values and raising the coefficient Λ_1. As a consequence, employing a regularizer promotes sparse representation while avoiding the problem of overfitting (underfitting). By following the procedures described above, the AE to be obtained first is trained. Once the training procedure is complete, the first AE hidden layer that corresponds to each 'X' is fed in the second AE as an input, and the bias and weights of the second AE learn by the decrease of the reconstruction error. By reducing the cross-entropy error produced at the layer of classification, the Deep Learning Neural Network (DLNN) fine-tunes using the backpropagation technique.

4. **Results and Discussion.** The proposed method's performance is evaluated using three datasets in this section: the SBU Kinect RGB-D video sequences [54], the UT interaction dataset [38], and the Weizmann dataset [12]. To evaluate the suggested method's performance across all datasets, leave one out cross validation is employed. On Windows 10, the proposed technique is performed using MATLAB R2016a and a Core i7 CPU.

4.1. **Datasets.** Each pair of continuous videos in the UT interaction dataset [38] includes six distinct ways of the interactions between two persons: Punch (PC), Hug (HG), Point (PT), Kick (KK), Shake Hands (SH), and Push (PS). Every pair includes ten videos, one from the parking lot and the other from a dynamic background. Exchange something (EX), Depart (DP), Approach (AP), kick, hug, push, shake hands and punch, are among the eight two-person interactions in the SBU Kinect interaction dataset [54]. The Weizmann (WZN) dataset [12] has ten known background actions: Run (RN), Pjump (PP), Jack (JK), Bend (BD), Walk (WK), Jump (JP), Side (SD), Skip (SP), Wave2 (W2) and Wave1 (W1).

4.2. **Experiment and analysis.** Through the three different techniques, we processed the videos used in an actual security surveillance project and compared the performance of the various context modeling techniques. Figure 6 shows the sample frames and the corresponding frames from the three methods when test with long video: frame difference, approximate median, a mixture of Gaussians. The first frame is the original frame, the second is the background frame, and the third is the foreground frame detected by each method. We checked the methods with changed parameters in different scenarios to decide which method to process the experiment data is the most effective. The results of evaluating a long-time video (600 frames) are shown in Figure 6 as follows. Figure 6(a) displays the effects of the frame difference algorithm's context subtraction. This technique has the highest data processing speed, but a significant flow of the technique is that a person must constantly be moving. It becomes part of the background if a person remains still for more than a frame duration (1/fps). The history subtraction effects of the approximate median algorithm are shown in Figure 6(b). This technique completes a much better job of separating an individual from the context, and the pace of processing is also rapid. This is because a long history of the visual scene combines the more slowly adapting context, producing roughly the same result as if we had buffered and processed N frames. Behind the guy, we can also see some trails.

This is because history is changed at a reasonably high rate (30 fps). The frame rate is possibly lower in a real application (15 fps). The effects of the background subtraction of the Gaussian algorithm mixture are shown in Figure 6(c). Compared to the above two methods, the MoG is very slow to deal with video data, but separating individuals and



(a) Frame difference (b) Approximate median (c) Mixture of Gaussians

FIGURE 6. Sample frames and the corresponding frames from the three methods when test with long video: frame difference, approximate median, a mixture of Gaussians. The first frame is the original frame, the second is the background frame, and the third is the foreground frame detected by each method.

suppressing background noise is fine. However, there are some stages where the technique breaks down, allowing much of the background to seep into the foreground. These points lead to relatively fast illumination shifts. The results of evaluating a short time video (60 frames) are shown in Figure 6. We also infer that Figure 6(a) indicates that the FD approach is substantially worse than other approaches when a short video is checked. There is also the issue of confusing the foreground with the past. Figure 6(b) shows that the AMF approach is not as good as MoG, but it produces good results with extremely fast implementation. Since it adapts slowly to an apparent shift in meaning, to learn the new history, AMF requires several frames. Hence, in the outcome shown in Figure 6(b), we can see obvious tracks. The best results are obtained by the MoG method, as shown in Figure 6(c). The multimodal features of the MoG can be attributed to this. However, the disadvantages of MoG are still clear. It is computer-intensive and involves careful tuning of the parameters. The method does not function anymore when the threshold is adjusted from 0.25 to 0.5. It shows the results of testing a short time video (60 frames) as follows. We need to evaluate the individual target more precisely when the moving target is detected from a stable and complex background. To locate each human body, we used a rectangular box. We can see some parts of continuous regions in the foreground image that correspond to nobody's portions from the binary image. Usually, it is caused by light changes and interference with equipment. Processing using the method of mathematical morphology is a more satisfactory process. We may then obtain a more accurate moving area of the human body.

We need to evaluate the individual target more precisely when the moving target is detected from a stable and complex background. To locate each human body, we used a rectangular box. We can see some parts of continuous regions in the foreground image that correspond to nobody's portions from the binary image. Usually, it is caused by light changes and interference with equipment. Processing using the method of mathematical morphology is a more satisfactory process. We may then obtain a more accurate moving area of the human body. It is possible to take advantage of those assumptions to construct a shadow classifier. A threshold on the difference is executed on the channel. The threshold on the absolute difference shows better results on channel H. With our proposed process, we checked the video used in the real surveillance project. The 60 frames of video were processed in conjunction with the context subtraction method referred to in the last chapter. After applying the method of mathematical morphology and region filling on the detected moving region, we can monitor. Figure 7 shows a background environment model and an image frame with a problematic shadow. The background subtraction operator detects the shadow as a part of the object. Figure 8 shows an example of shadow behavior in HSV color space for a sample background and image frame. We infer that

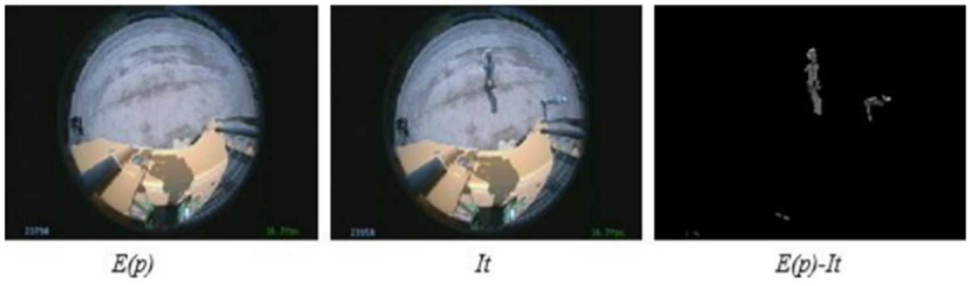


FIGURE 7. A background environment model and an image frame with a problematic shadow. The background subtraction operator detects the shadow as a part of the object.

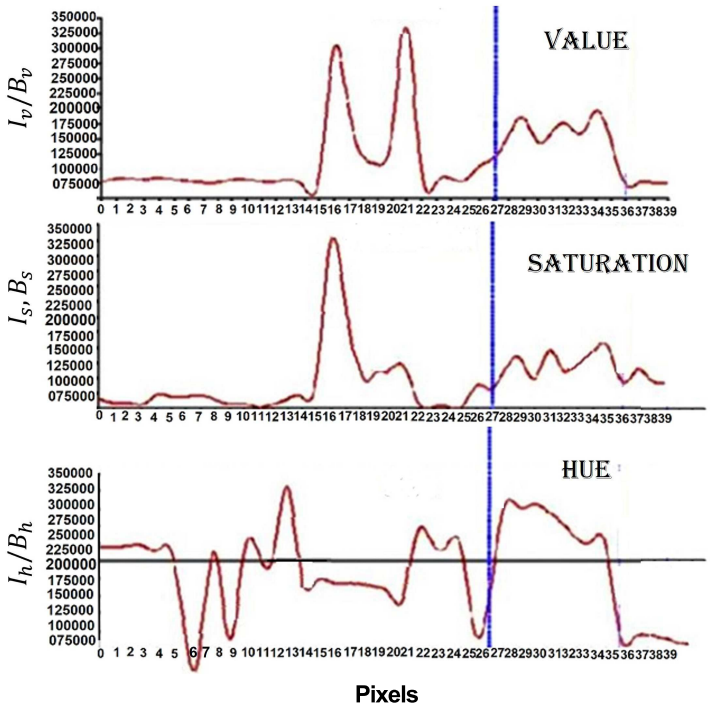


FIGURE 8. An example of shadow behavior in HSV color space for a sample background and image frame. Analyzing the relations between pixels in different channels is possible to obtain a set of classification rules for shadow pixels (marked in the illustration by vertical dotted lines).

TABLE 1. Accuracy performance on different datasets in comparison with the state-of-the-art methods

Method	SBU interaction dataset	UT interaction dataset	Weizmann dataset
Huynh-The et al. [16]	90	91	91
Li et al. [29]	84	90	92
Ji et al. [19]	87	91	93
Chebi et al. [28]	87	91	93
DWT ($a = 0.2$)	80	81	88
DWT + Harris ($a = 0.2$)	91	91	90
Our deep learning method	97.09	98.02	96.17

analyzing the relations between pixels in different channels is possible to obtain a set of classification rules for shadow pixels (marked in the illustration by vertical dotted lines). Table 1 determines the accuracy performance on different datasets in comparison with the state-of-the-art methods. The results infer our method's better performance with an accuracy of 97.09% on the SBU interaction dataset, 98.02% on the UT interaction dataset, and 96.17% on the Weizmann dataset. Table 2 gives the statistical values of our method, determining the accuracy of our proposed method. The values are high for the various parameters. This proves our method's superior performance.

The outcomes shown in the graph in Figure 8 are relatively good. However, there are still some issues; as shown in the figure, we can easily find that the algorithm for shadow

TABLE 2. Statistical values of our method determining the accuracy of our proposed method. Note that MCC stands for Mathews Coerrelation Coefficient.

Specificity	Accuracy	Dice	F-Measure	Jaccard	MCC	Precision	Sensitivity
98.02	97.09	96.17	96.17	92.63	93.83	96.76	95.59

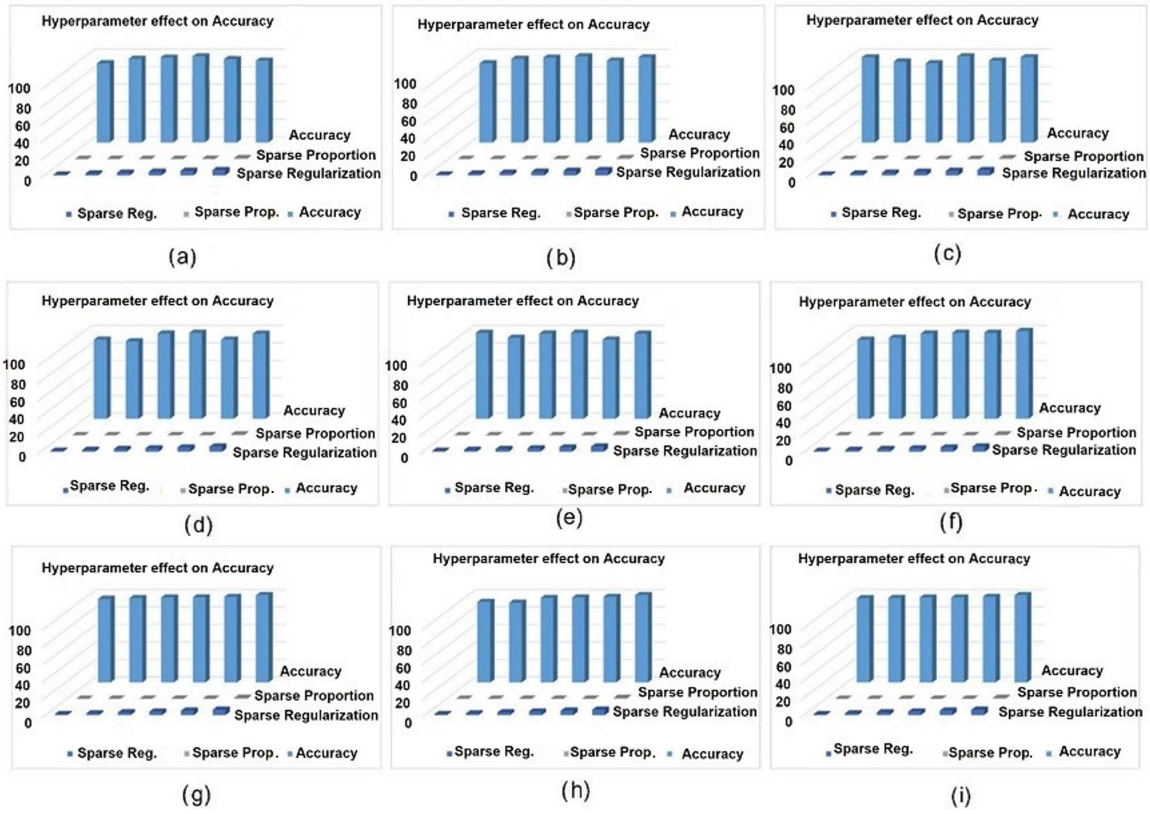


FIGURE 9. Effect of sparse regularization ($\Lambda 2$), sparse proportion and L2 regularization ($\Lambda 1$) on the percentage of accuracy while evaluating for various number of hidden nodes of a DLNN having two AE's: (a) $\Lambda 1 = 0, 250$; (b) $\Lambda 1 = 0.001, 250$; (c) $\Lambda 1 = 0.01, 250$; (d) $\Lambda 1 = 0, 100$; (e) $\Lambda 1 = 0.001, 100$; (f) $\Lambda 1 = 0.01, 100$; (g) $\Lambda 1 = 0, 50$; (h) $\Lambda 1 = 0.001, 50$; (i) $\Lambda 1 = 0.01, 50$

removal is not good enough to decrease shadow impact. We do need to do further work on the issue. PSO and Parzen particle filter algorithms are widely used in other fields of study, such as management engineering, and industry planning. Musa and Adamu [77] first suggested a moving target tracking system based on the PSO and particle filter algorithm in our research community. We have used this methodology in our research to track people more effectively. To detect and track a static and moving object that helps us to solve the following problems, PSO and Parzen filter algorithm are suggested based on template matching. Figure 14 shows the detected results using our algorithm. Note the bounding box for the two persons in the 4th block of the figure. As it can be seen, the human detected is very clear with our method.

This helps us to solve the following problems; PSO and Parzen filter algorithms are suggested based on template matching. Two sparse AEs are separately trained to generate weights taken initially for the deep learning neural network. The encoder side weights for

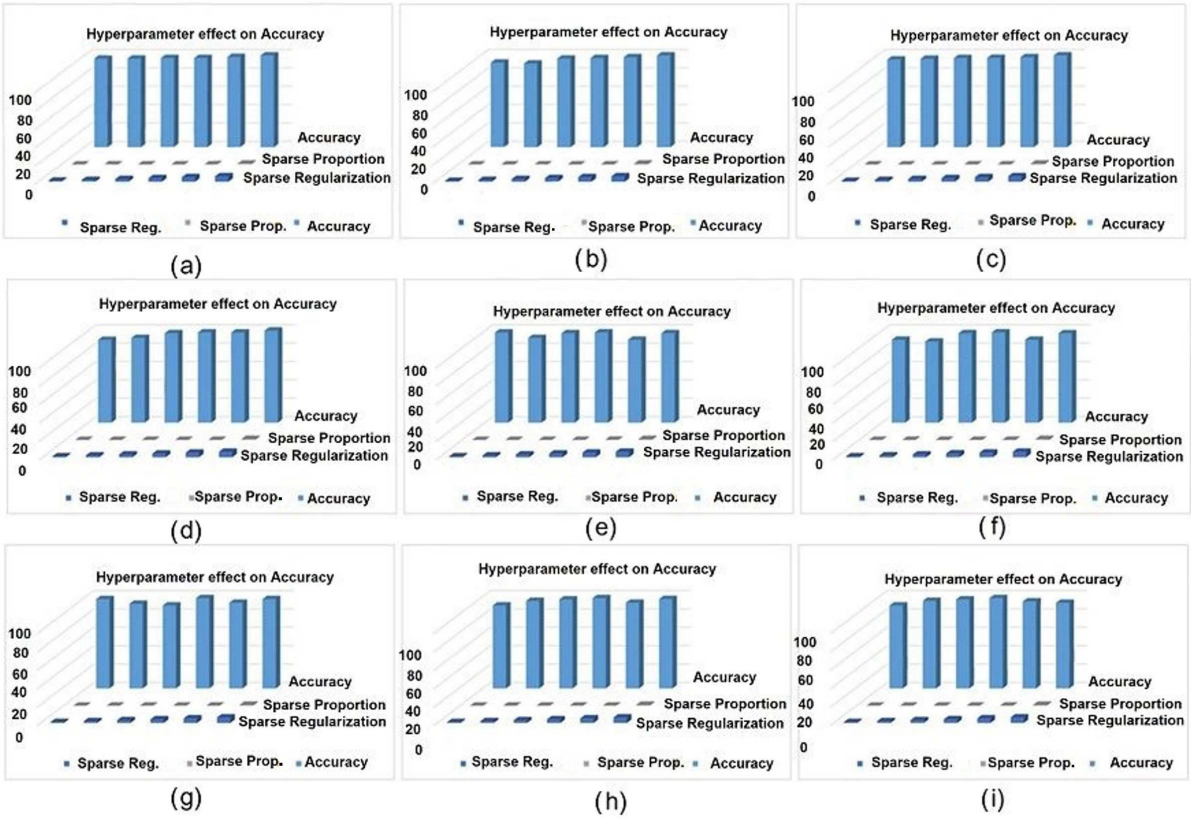


FIGURE 10. Effect of sparse regularization ($\Lambda 2$), sparse proportion and L2 regularization ($\Lambda 1$) on the percentage of accuracy while evaluating for various number of hidden nodes of a DLNN having single AE: (a) $\Lambda 1 = 0, 250$; (b) $\Lambda 1 = 0.001, 250$; (c) $\Lambda 1 = 0.01, 250$; (d) $\Lambda 1 = 0, 100$; (e) $\Lambda 1 = 0.001, 100$; (f) $\Lambda 1 = 0.01, 100$; (g) $\Lambda 1 = 0, 50$; (h) $\Lambda 1 = 0.001, 50$; (i) $\Lambda 1 = 0.01, 50$

the first AE are established using weights present between the first hidden node and the input layer of the DLNN. Similarly, the DLNN's first weights and second hidden nodes are used to set the side weights of the second AE (encoder). The AE uses the logsig transfer function since it outperforms satlin and purelin. Trainscg and msesparse are the loss function and the training technique used to learn the AE, respectively, while softmax and cross-entropy are the loss function and activation function used in the DLNN neurons output. The number of DLNN output neurons in the database changes in response to the number of target actions. Both the DLNN and autoencoders are trained for 500 and 200 epochs, respectively. On the other hand, the inertia weight is adjusted regularly, and the acceleration constants are set to 2. To minimize the gap between the reconstruction of input feature values and their using AE, hyper-parameters such as sparsity regularization, L2 weight regularization, and sparsity percentage are taken. The assessment of various combined three parameters is given as an input into either a single or a double layer network with a configurable hidden number of nodes for hyper-parameters to maximization. With two stacked AEs and a single AE, Figures 9 and 10 illustrate how hyper-parameters influence DLNN accuracy.

Sparse proportion is set to 0.02 to 0.1 with a 0.02 step, and sparse regularization is set to 1 to 6 with a 1 step. One of three potential options for the L2 regularization coefficient is 0.001, 0.01 or 0. The greatest performance is noted when sparse percentage, L2 regularization, and sparse regularization are set to 0.02, 0.01, and 1, respectively. Several

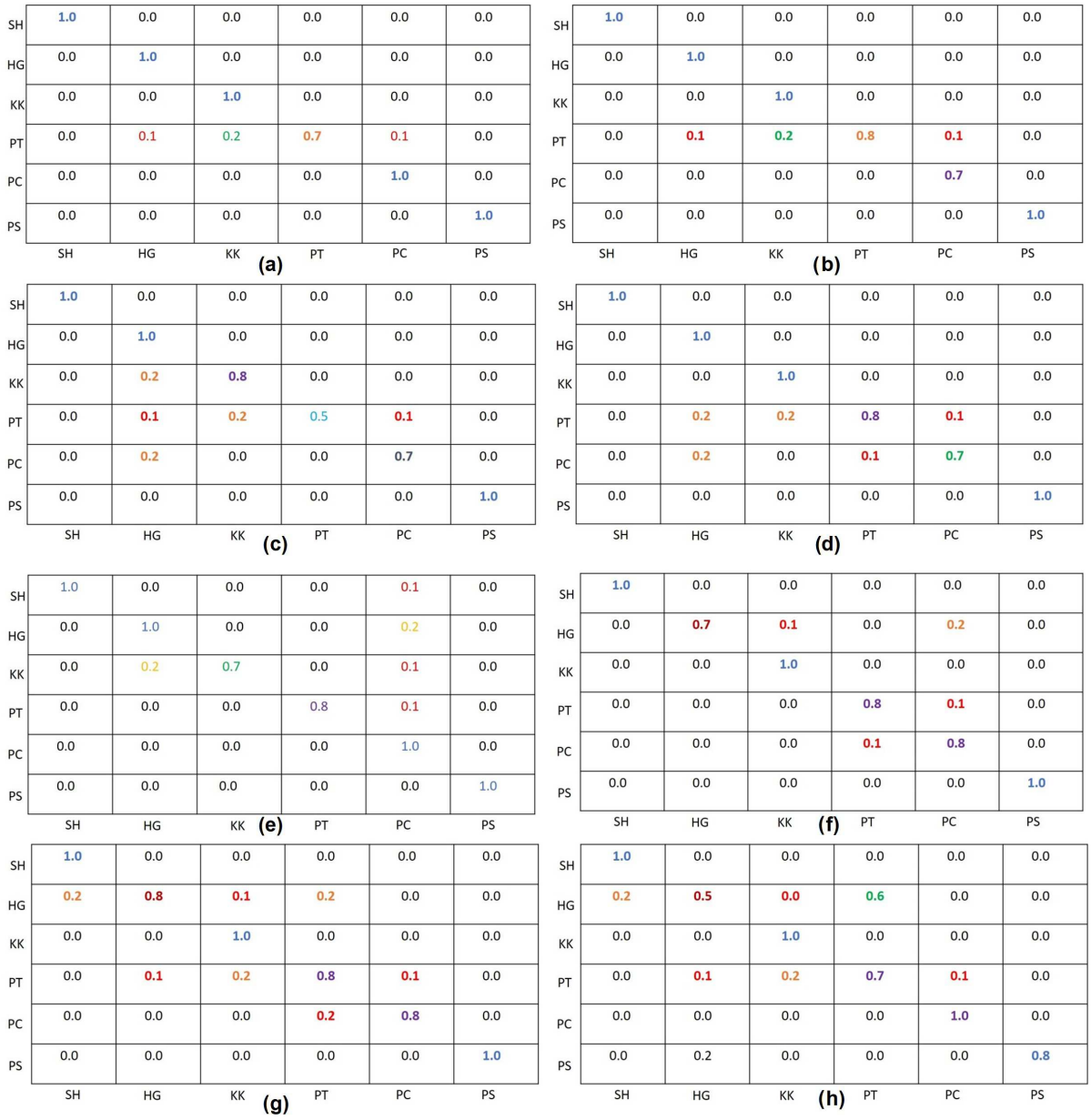


FIGURE 11. Confusion matrix on UT interaction dataset: (a) Set 1: With PSO ($a = 0$); (b) set 1: With PSO ($a = 0.2$); (c) set 1: With PSO ($a = 0.5$); (d) set 1: Without PSO; (e) set 2: With PSO ($a = 0$); (f) set 2: With PSO ($a = 0.2$); (g) set 2: With PSO ($a = 0.5$); (h) set 2: Without PSO

experiments conducted infer that 50 and 100 neurons are sufficient in the second and first hidden layers, respectively, on the basics of computational complexity and accuracy. In the performance analysis taken on the UT interaction dataset, there are two sets in this dataset, having a dynamic background and the other with a static background, GMM and background removal are done independently on set 1 and set 2. There are ten sequences of videos in total for every activity. In each interaction, one of the sequences is used in the testing, and the remained sequences are used in the training process. Figure 11 depicts the confusion matrix of the UT interaction datasets. The recognition accuracies on the UT interaction set for many acts such as a hug, shake hand, and push are excellent. The punch and point actions are the most puzzling because the actions in both involve only a little variation in the movement of hand. Furthermore, the foreground image generated by

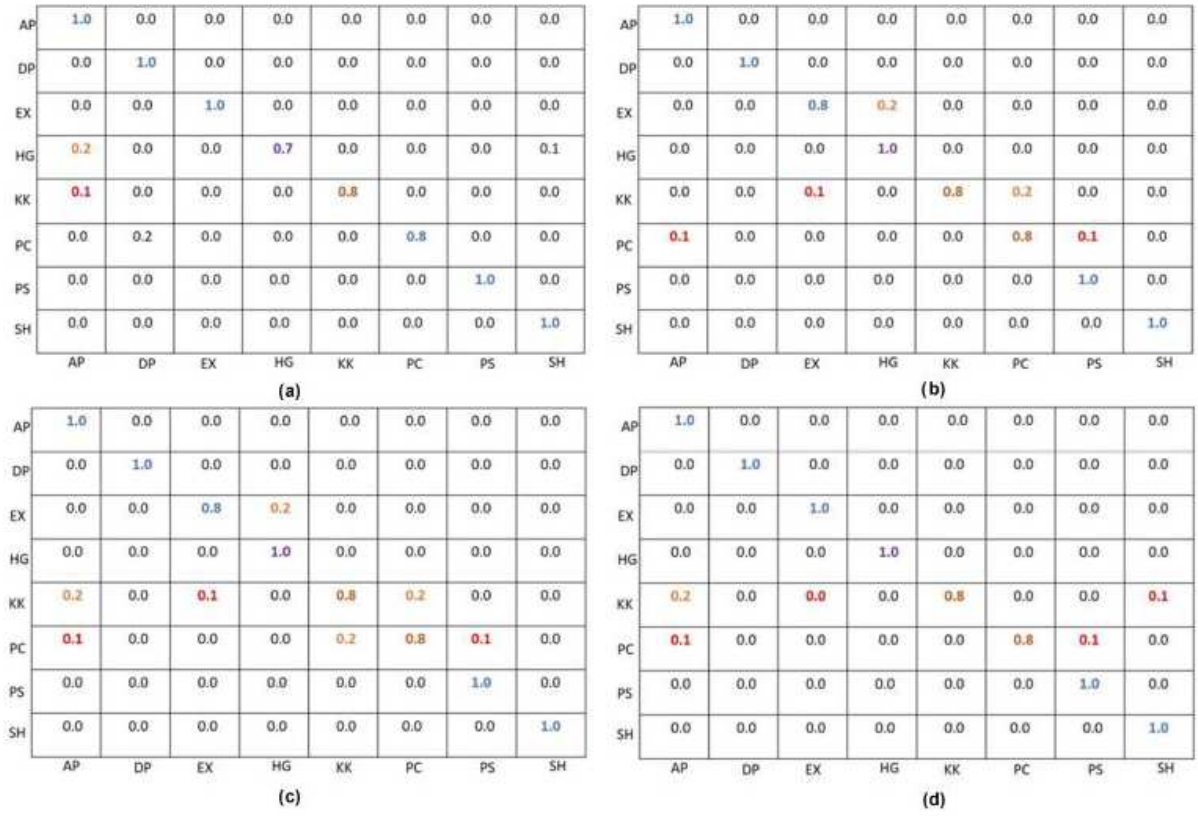


FIGURE 12. Confusion matrix on SBU interaction dataset: (a) With PSO ($a = 0$); (b) with PSO ($a = 0.2$); (c) with PSO ($a = 0.5$); (d) without PSO

UT interaction set 2 is less clear than that produced by set 1 of UT interaction due to the occlusion and covered background. The movement patterns associated with interactions such as punch, kick, and push are similar at some points in the UT interaction dataset, indicating that they were mislabeled. Figure 12 depicts the SBU interaction dataset's confusion matrix. The diagonal elements are accurately identified, while the remaining values are misclassified. The push, interactions apart, depart, and shake hand interactions prove to work perfectly with a 100% classification rate. It involves whole-body movement rather than simply leg movement. The action punch is often mislabeled as a push since the postures of both encounters are so similar. Unlike the SBU and UT interaction datasets, the Weizmann dataset is designed for single person human interactions. The action sequence of the foreground mask is clearly defined and utilized to go for further process. On the set 1 and set 2, the best classification accuracy is 98.02%. The proposed method accuracy rate is compared to the reported results by the current state of the art methods in Table 1. The results obtained are compared to state-of-the-art methods. Set 2 has a cluttered background with trees moving, a spectator, and a camera jitter, among other challenges. Method [32] utilizes the computation of gradient histogram and spatiotemporal interest points for generating the feature. Our proposed method significantly outperforms the other techniques with fewer features obtained from a single spatiotemporal volume. The SBU interaction dataset includes both human-to-object and human-to-human interaction. Because the series' depth-based images are given directly in this instance, thresholding is used to get the foreground image.

The Weizmann dataset confusion matrix is shown in Figure 13. According to the graph, classification accuracy for specific movements like side, jack, bend, skip, and wave reaches

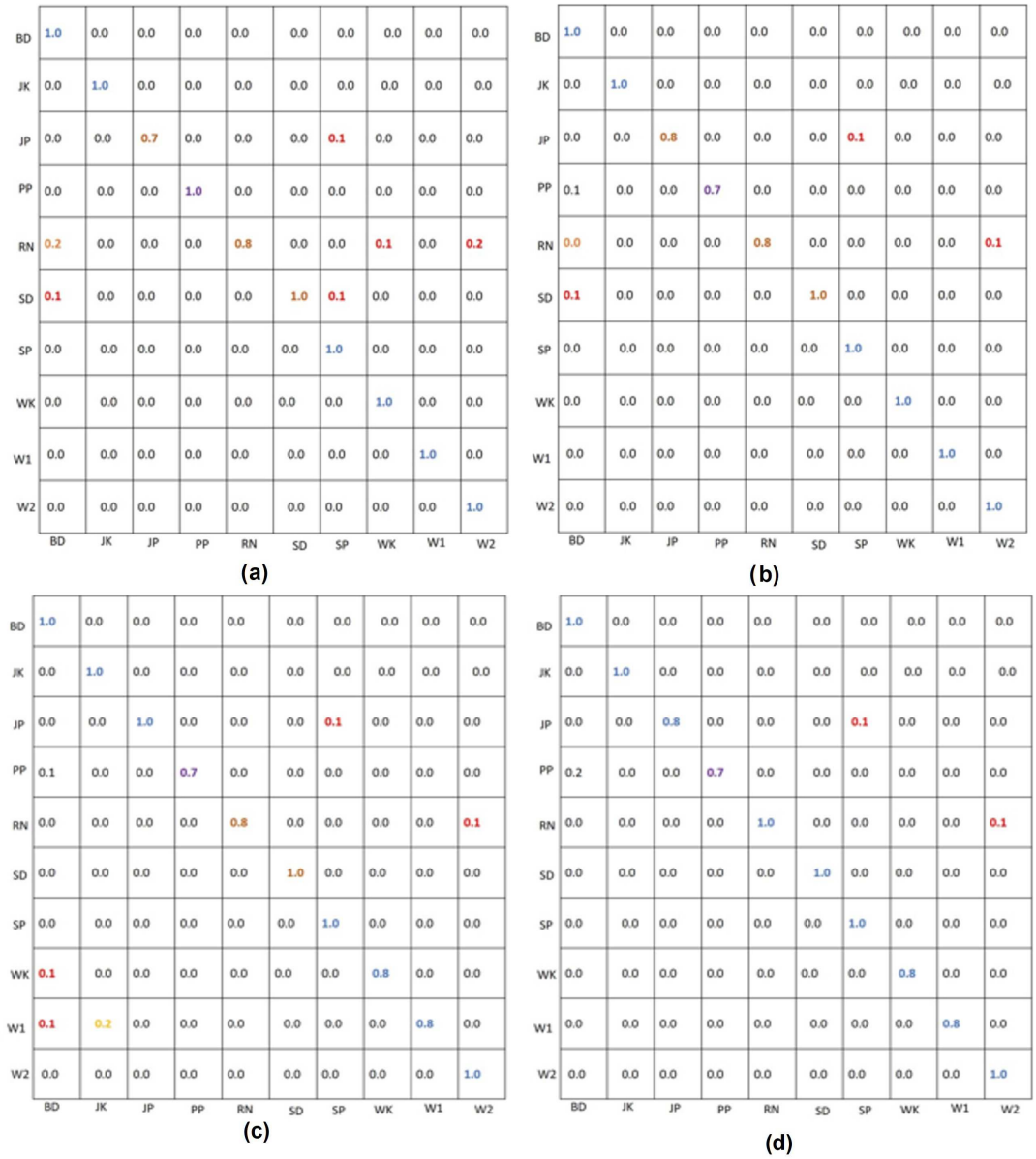


FIGURE 13. Confusion matrix on Weizmann dataset: (a) With PSO ($a = 0$); (b) with PSO ($a = 0.2$); (c) with PSO ($a = 0.5$); (d) without PSO

almost 100%. It can be seen that some misclassification happens in jump, run, stroll, and wave. It is tough to recognize these activities apart since they all have comparable structural and kinematic characteristics. As a consequence, specific instances of action are more likely than others to be identified. Thus the suggested strategy outperforms the methods described in [2,55]. When compared to the method used in [4] the proposed study yields comparable results. Note the bounding box for the two persons in the 4th block of the figure. As it can be seen, the human detected is very clear with our method. The outcome of the detection process shows that the human position can be recognized by our proposed system, as shown in Figure 8. In addition, the output indicates that the target is more precise, and the position can be easily identified. In addition, it is possible to classify the human position using the PSO algorithm and Parzen particle filter and track multiple human movements for each frame, as shown in Figure 15. Moreover, compared to the continuous detection method, our proposed method can monitor human

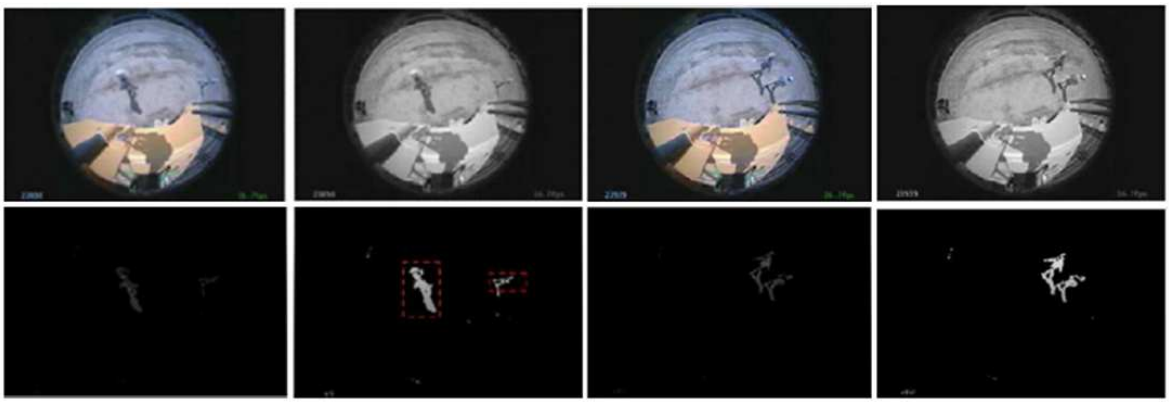


FIGURE 14. The detected results using our algorithm. Note the bounding box for the two persons in the 4th block of the figure. As it can be seen the humans detected are very clear with our method.

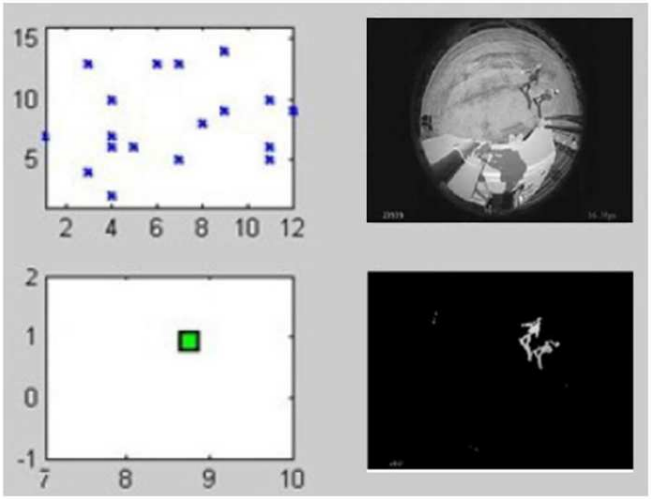


FIGURE 15. Tracking result based on our proposed method

activities before or after the overlap between objects in Figure 8 – An example of shadow behavior in HSV color space for a sample background image frame. The relations can be analyzed in different channels to obtain a set of classification rules for the shadow pixels and are illustrated by the vertical dotted lines in Figure 8. And when all of these objects are overlapped, it is capable of discriminating against two distinct object movements.

5. **Conclusions.** This article presents a deep learning framework for human action recognition. It involves the identification, segmentation, and tracking in smart video surveillance monitoring around a huge truck. We processed videos taken by a fisheye camera (FOV) which can cover more than 180 degrees of FOV, so we can use fewer cameras to track the entire environment. To decrease computational complexity, the characteristics of video frame sequences are reduced using the particle swarm optimization detection method.

If the scene is still stationary, the detection of moving individuals is handled using the correlation tracking method. Finally, the efficacy of the approach is assessed using a deep learning neural network. Because the information in the autoencoders is trained independently, the information in the autoencoders is passed to the deep learning neural network to identify human actions. Based on the Hue Saturation Value color space, our

deep learning approach also conducts atomic morphological operations including shadow removal. Our technique can successfully track objects with low contrast and prominent characteristics against the background. We conclude that our method improves tracking stability and enhances the robustness of the tracking process based on the quality of the tracking data. Amongst the three distinct context modeling techniques evaluated on-ground video sequences in human body motion detection, the Gaussian mixture produces the best results but has the slowest processing speed. The good results and faster processing speed are given by an approximate median. To strengthen the robustness of our software against ambient noise, light changes and other factors of effect have also been taken into consideration. Many video data have been examined in this research, such as single and multiple human gestures with distinct directions. On the other hand, in the proposed process, our tracking module can track the human position in a stable manner. Our statistical results infer that our method performs better than all the current state-of-the-art methods by an increase in accuracy of 5%, 6% and 7% on the Weizmann dataset SBU interaction dataset and the UT interaction dataset respectively. In our future work, we will focus more on enhancing the processing speed.

Acknowledgement. This research is the result of a joint project with the Safety Device Unit of the KOMATSU Research Division and Yayasan Universiti Teknologi Petronas (YUTP), Cost Center 015LC0-281, Project title: Deep Learning Model of Masking Vision Based Panoramic View Understanding to Detect Non-safety Situations in Mining.

REFERENCES

[1] A. Lee, W. Chung, S.-P. Yong and J. Watada, Global thresholding for scene understanding towards autonomous drone navigation, *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol.23, no.5, pp.909-919, 2019.

[2] U. A. Usmani, J. Watada, J. Jaafar and A. Roy, Enhanced deep learning model for extraction of irregular objects in complex imaging, *The Computing Conference 2021*, The Science and Information (Sai) Organization Limited, United Kingdom, 2021.

[3] Y. Shin, M. Kim, K.-W. Pak and D. Kim, Practical methods of image data preprocessing for enhancing the performance of deep learning based road crack detection, *ICIC Express Letters, Part B: Applications*, vol.11, no.4, pp.373-379, 2020.

[4] Z. Xu, J. Watada and Z. B. Musa, Particle filter-based height estimation in human tracking, *The 5th International Conference on Genetic and Evolutionary Computing*, pp.385-388, DOI: 10.1109/ICGEC.2011.94, 2011.

[5] Z. B. Musa, J. Watada and H. Zhang, Multi-camera tracking method based on particle filtering, *2010 World Automation Congress*, 2010.

[6] A. Dasgupta, R. Hostanche, R. A. A. J. Ramsankaran, G. J.-P. Schumann, S. Grimaldi, V. R. N. Pauwels and J. P. Walker, A mutual information-based likelihood function for particle filter flood extent assimilation, *Water Resources Research*, vol.57, no.2, DOI: 10.1029/2020WR027859, 2021.

[7] W. Quan and J. Watada, Clustering and forecasting of region of interest by dividing screen into meshes in video frames, *2014 Joint the 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and the 15th International Symposium on Advanced Intelligent Systems (ISIS)*, pp.839-844, DOI: 10.1109/SCIS-ISIS.2014.7044896, 2014.

[8] Z. Musa, M. Z. Salleh, R. A. Bakar and J. Watada, GbLN-PSO and model based particle filter approach for tracking human movements in large view cases, *IEEE Trans. Circuits and Systems for Video Technology*, vol.26, no.8, pp.1433-1446, 2016.

[9] D. Sun and J. Watada, Detecting pedestrians and vehicles in traffic scene based on boosted HOG features and SVM, *2015 IEEE the 9th International Symposium on Intelligent Signal Processing (WISP) Proceedings*, DOI: 10.1109/WISP.2015.7139161, 2015.

[10] M. Gong, J. Liu, H. Li, Q. Cai and L. Su, A multiobjective sparse feature learning model for deep neural networks, *IEEE Trans. Neural Networks and Learning Systems*, vol.26, no.12, pp.3263-3277, DOI: 10.1109/TNNLS.2015.2469673, 2015.

[11] M. Gong, J. Zhao, J. Liu, Q. Miao and L. Jiao, Change detection in synthetic aperture radar images based on deep neural networks, *IEEE Trans. Neural Networks and Learning Systems*, vol.27, no.1, pp.125-138, 2015.
[12] L. Gorelick, M. Blank, E. Shechtman, M. Irani and R. Basri, Actions as spatio-temporal shapes, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.8, 2007.
[13] D. M. Gavrila, The visual analysis of human movement: A survey, *Computer Vision and Image Understanding*, vol.73, no.1, pp.82-98, 1999.
[14] I. Haritaoglu, D. Harwood and L. Davis, W4: Real-time surveillance of people and their activities, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.22, no.8, pp.809-830, 2000.
[15] J. Y. Ma, F. R. Jie and Y. J. Hu, Moving target detection method based on improved Gaussian mixture model, *Proc. of SPIE 10420, the 9th International Conference on Digital Image Processing (ICDIP2017)*, DOI: 10.1117/12.2282506, 2017.
[16] L. Maddalena and A. Petrosino, A self-organizing approach to background subtraction for visual surveillance applications, *IEEE Trans. Image Processing*, vol.17, no.7, pp.1168-1177, 2008.
[17] J. Berclaz, F. Fleuret, E. Turetken and P. Fua, Multiple object tracking using k-shortest paths optimization, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.33, no.9, pp.1806-1819, 2011.
[18] D. Feng, M. Liang and G. Wang, A traffic anomaly detection and identification approach based on multi-instance learning, *The 2nd International Conference on Computing and Data Science*, pp.1-7, 2021.
[19] Y. Ji, G. Ye and H. Cheng, Interactive body part contrast mining for human interaction recognition, *IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pp.1-6, 2014.
[20] J. C. Tai, S. T. Tseng, C. P. Lin and K. T. Song, Real-time image tracking for automatic traffic monitoring and enforcement applications, *Image and Vision Computing*, vol.22, no.6, pp.485-501, 2004.
[21] C.-J. Pai, H.-R. Tyan, Y.-M. Liang, H.-Y. M. Liao and S.-W. Chen, Pedestrian detection and tracking at crossroads, *Pattern Recognition*, vol.37, no.5, pp.1025-1034, 2004.
[22] P. Marcha, SAVE-U: Sensors and system architecture for vulnerable road users protection, *Proc. of the 3rd Concertation Meeting*, Bruxelles, Belgium, 2004.
[23] D. Mayer, H. Denzler and Nieman, Model based extraction of articulated objects in image sequences for gait analysis, *Proc. of International Conference of Image Processing*, pp.78-81, 1998.
[24] Stauffer and W. E. L. Grimson, Learning patterns of activity using real-time tracking, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.22, no.8, pp.747-757, 2000.
[25] R. Alzahabi and M. S. Cain, Ensemble perception during multiple-object tracking, *Attention, Perception, & Psychophysics*, vol.83, no.3, pp.1263-1274, 2021.
[26] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld and H. Wechsler, Tracking groups of people, *Computer Vision and Image Understanding*, vol.80, no.1, pp.42-56, 2000.
[27] A. Abdelhalim, M. Abbas, B. B. Kotha and A. Wicks, A framework for realtime traffic trajectory tracking, speed estimation, and driver behavior calibration at urban intersections using virtual traffic lanes, *arXiv Preprint*, arXiv: 2106.09932, 2021.
[28] H. Chebi, D. Acheli and M. Kesraoui, Automatic shadow elimination in a highdensity scene, *International Journal of Intelligent Systems Design and Computing*, vol.2, nos.3-4, pp.224-237, 2018.
[29] C. Stauffer and W. E. L. Grimson, Learning patterns of activity using real-time tracking, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.22, no.8, pp.747-757, 2000.
[30] A. J. Lipton, H. Fujiyoshi and R. S. Patil, Moving target classification and tracking from real-time video, *Proc. of the 4th IEEE Workshop on Applications of Computer Vision (WACV'98)*, pp.8-14, DOI: 10.1109/ACV.1998.732851, 1998.
[31] B. Jung and G. S. Sukhatme, Detecting moving objects using a single camera on a mobile robot in an outdoor environment, *The 8th Conference on Intelligent Autonomous Systems*, pp.980-987, 2004.
[32] B. Liao, J. Hu and R. O. Gilmore, Optical flow estimation combining with illumination adjustment and edge refinement in livestock UAV videos, *Computers and Electronics in Agriculture*, vol.180, DOI: 10.1016/j.compag.2020.105910, 2021.
[33] R. Cutler and L. S. Davis, Robust real-time periodic motion detection, analysis, and applications, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.22, no.8, pp.781-796, 2000.
[34] W. Xing, Y. Yang, S. Zhang, Q. Yu and L. Wang, NoisyOTNet: A robust real-time vehicle tracking model for traffic surveillance, *IEEE Trans. Circuits and Systems for Video Technology*, 2021.
[35] L. Wang, W. M. Hu and T. N. Tan, A survey of visual analysis of human motion, *Chinese Journal of Computers*, vol.25, no.3, pp.225-237, 2002.

[36] J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol.8, no.6, pp.679-698, 1986.
[37] S. Smith and J. Brady, SUSUN – A new approach to low level image processing, *International Journal of Computer Vision*, vol.23, no.1, pp.45-78, 1997.
[38] M. Kass, A. Witkin and D. Terzopoulos, Snakes: Active contour models, *International Journal of Computer Vision*, vol.1, no.4, pp.321-331, 1988.
[39] X. Zhang, Adaptive path planning control of snake-like robot based on reinforcement tracking learning, *The 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*, pp.1091-1095, 2021.
[40] S. Osher and J. A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational Physics*, vol.79, no.1, pp.12-49, 1988.
[41] R. Malladi, J. A. Sethian and B. C. Vemuri, Shape modeling with front propagation: A level set approach, *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol.17, no.2, pp.158-175, 1995.
[42] H. C. Shin, M. R. Orton, D. J. Collins, S. J. Doran and M. O. Leach, Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4D patient data, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.35, no.8, pp.1930-1943, 2012.
[43] M. A. Khan, S. Kadry, P. Parwekar, R. Damaševicius, A. Mehmood, J. A. Khan and S. R. Naqvi, Human gait analysis for osteoarthritis prediction: A framework of deep learning and kernel extreme learning machine, *Complex & Intelligent Systems*, pp.1-19, 2021.
[44] A. Yilmaz, O. Javed and M. Shah, Object tracking: A survey, *ACM Computing Surveys (CSUR)*, vol.38, no.13, DOI: 10.1145/1177352.1177355, 2006.
[45] S. L. Dockstader and A. M. Tekalp, On the tracking of articulated and occluded video object motion, *Real-Time Imaging*, vol.7, no.5, pp.415-432, 2001.
[46] A. Yilmaz, L. Xin and M. Shah, Contour based object tracking with occlusion handling in video acquired using mobile cameras, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.26, no.11, pp.1531-1536, 2004.
[47] J. Gao, Self occlusion immune video tracking of objects in cluttered environments, *Proc. of the IEEE Conference on Advanced Video and Signal Based Surveillance*, pp.79-84, DOI: 10.1109/AVSS.2003.1217905, 2003.
[48] A. Banerjee, P. Burlina and J. Broadwater, Hyper spectral video for illumination invariant tracking, *The 1st Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS'09)*, DOI: 10.1109/WHISPERS.2009.5289103, 2009.
[49] B. Xiao, Q. Lin and Y. Chen, A vision based method for automatic tracking of construction machines at nighttime based on deep learning illumination enhancement, *Automation in Construction*, vol.127, DOI: 10.1016/j.autcon.2021.103721, 2021.
[50] A. Moussavi, S. Mißbach, C. S. Ferrel, H. Ghasemipour, K. Kötz, C. Drummer, R. Behr, W. H. Zimmermann and S. Boretius, Comparison of cine and realtime cardiac MRI in rhesus macaques, *Scientific Reports*, vol.11, no.1, pp.1-12, 2021.
[51] B. Xue, M. Zhang and W. N. Browne, Particle swarm optimization for feature selection in classification: A multi-objective approach, *IEEE Trans. Cybernetics*, vol.43, no.6, pp.1656-1671, 2012.
[52] I. Ryuichi, Edge preservation noise reduction scheme and its application to galloping images, *ITE Technical Report*, vol.26, no.60, pp.7-12, 2002.
[53] C. Tsai and K. Song, Dynamic visual tracking control of a mobile robot with image noise and occlusion robustness, *Image and Vision Computing*, vol.27, no.8, pp.1007-1022, 2009.
[54] R. Kitahara, T. Nakamura, A. Katayama and T. Yasuno, Real time rectangle tracking method for geometric correction on mobile terminals, *Technical Report of IEICE*, vol.106, no.351, 2006 (in Japanese).
[55] B. N. Hahn, Motion compensation strategies in tomography, *Time Dependent Problems in Imaging and Parameter Identification*, pp.51-83, 2021.
[56] H. Huang, F. Zhong and X. Qin, Pixel-wise weighted region-based 3D object tracking using contour constraints, *IEEE Trans. Visualization and Computer Graphics*, 2021.
[57] A. Cavallaro, O. Steiger and T. Ebrahimi, Tracking video objects in cluttered background, *IEEE Trans. Circuits and Systems for Video Technology*, vol.15, no.4, pp.575-584, 2005.
[58] H. S. J. McKenna, S. Jabri, Z. Duric and H. Wechsler, Tracking interacting people, *IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.
[59] D. M. Lyons and D. F. Hsu, Combining multiple scoring systems for target tracking using rank score characteristics, *Information Fusion*, vol.10, no.2, pp.124-136, 2009.

[60] P. Reisman, O. Mano, S. Avidan and A. Shashua, Crowd detection in video sequences, *Proc. of the Symposium on Intelligent Vehicles*, pp.66-71, 2004.

[61] C. S. Regazzoni and G. L. Foresti, Guest editorial: Video processing and communications in real-time surveillance systems, *Real-Time Imaging*, vol.7, no.5, pp.381-388, 2001.

[62] M. Greiffenhagen, V. Ramesh and D. Comaniciu, Statistical modeling and performance characterization of a real-time dual camera surveillance system, *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol.2, pp.335-342, 2000.

[63] G. Welch, B. D. Allen, A. Ilie and G. Bishop, Measurement sample time optimization for human motion tracking/capture systems, *Proc. of Trends and Issues in Tracking for Virtual Environments*, Workshop at the IEEE Virtual Reality, 2007.

[64] M. Hoch and P. C. Litwinowicz, A semi-automatic system for edge tracking with snakes, *The Visual Computer*, vol.12, pp.75-83, DOI: 10.1007/BF01782106, 1996.

[65] Q. Gao, A. Parslow and M. Tan, Object motion detection based on perceptual edge tracking, *The 2nd International Workshop on Digital and Computational Video (DCV'01)*, 2001.

[66] E. D. Dickmanns and V. Graefe, Dynamic monocular machine vision, *Machine Vision and Applications*, vol.1, no.4, pp.223-240, 1988.

[67] G. Zheng and Y. Xu, Efficient face detection and tracking in video sequences based on deep learning, *Information Sciences*, vol.568, pp.265-285, 2021.

[68] H. Stern and B. Efros, Adaptive color space switching for face tracking in multi-colored lighting environments, *The 5th IEEE International Conference on Automatic Face and Gesture Recognition*, pp.249-254, 2002.

[69] A. Naiman, Color spaces and color contrast, *The Visual Computer*, vol.1, no.3, pp.194-201, 1985.

[70] M. M. Aznaveh, H. Mirzaei, E. Roshan and M. H. Saraee, A new and improved skin detection method using mixed color space, *Human-Computer Systems Interaction*, pp.471-480, 2009.

[71] C.-F. Lin, S.-W. Yeh, Y.-Y. Chien, T.-I. Peng, J.-H. Wang and S.-H. Chang, A HHT-based time frequency analysis scheme for clinical alcoholic EEG signals, *WSEAS Trans. Biology and Biomedicine*, vol.5, no.10, pp.249-260, 2009.

[72] W. Wang and Y. H. Liur, Motion features, *Proc. of the 9th WSEAS International Conference on Robotics, Control and Manufacturing Technology*, pp.67-72, 2009.

[73] H.-F. Ng and Y.-W. Chu, Illumination invariant color model for image matching and object recognition, *The 8th International Conference on Intelligent Systems Design and Applications*, vol.1, pp.95-99, 2008.

[74] A. Nawrat and K. Jedrasiak, Seth system spatio-temporal object tracking using combined color and motion features, *Proc. of the 9th WSEAS International Conference on Robotics, Control and Manufacturing Technology*, pp.67-72, 2009.

[75] M. Kristan, J. Pers, S. Kovacic and A. Leonardis, A local-motion-based probabilistic model for visual tracking, *Pattern Recognition*, vol.42, no.9, pp.2160-2168, 2009.

[76] M. Gavilán, D. Balcones, O. Marcos, D. F. Llorca, M. A. Sotelo, I. Parra, M. Ocaña, P. Aliseda, P. Yarza and A. Amírola, Adaptive road crack detection system by pavement classification, *Sensors*, vol.11, no.10, pp.9628-9657, 2011.

[77] H. Musa and S. S. Adamu, Enhanced PSO based multi-objective distributed generation placement and sizing for power loss reduction and voltage stability index improvement, *IEEE Energytech*, pp.1-6, 2013.

Author Biography

Usman Ahmad Usmani was born in Aligarh, India, April 1993. He is currently a Ph.D. Computer Science student at the Universiti Teknologi Petronas, Malaysia. He has worked as a research assistant at IIT Kanpur and as a researcher in Massey University, New Zealand. He has built up a social network named Zamber that has been published in around 14 national newspapers. His areas of research interests are artificial intelligence, computer vision, computer security, wearable sensors, cloud computing.

Junzo Watada (Life Senior Member, IEEE) received the B.Sci. and M.Sci. degrees in electrical engineering from Osaka City University, Osaka, Japan, and the Ph.D. degree from Osaka Prefecture University, Sakai, Japan. After retiring Waseda University, he contributed, as a full Professor, the Department of Computer and Information Sciences, Universiti Technologi Petronas, Malaysia, and a Professor Emeritus with Waseda University, Japan. His research interests include big data analytics, soft computing, image processing systems to track human behaviours and understand pictures and videos, knowledge engineering, and management engineering.

Jafreezal Jaafar is an Associate Professor and former Head of the Computer and Information Sciences Department at Universiti Teknologi PETRONAS, Malaysia. He holds a Ph.D. from University of Edinburgh, UK (2009). His main research areas include big data analytics, soft computing and software engineering. He has secured a number of research projects from the industry and government agencies. Based on his publication track records he had been appointed as the Chief Editor and reviewer for several journals, and also the Chair, Technical Chair and committee for several International Conferences. He is also active in IEEE Computer society, Malaysia Chapter and has been appointed as the Executive Committee for 2016 and 2017.

Izzatdin Abdul Aziz is currently heading the Center for Research in Data Science (CeRDaS) at The Universiti Teknologi PETRONAS (UTP), where he focuses in solving complex upstream Oil and Gas (O&G) industry problems from the view point of data analytics, machine learning, big data and AI. Dr. Izzatdin currently serves as the deputy head of the Computer and Information Sciences Department in UTP. He obtained his Ph.D. in Information Technology from Deakin University, Australia working in the domain of hydrocarbon exploration and cloud computing. He is working closely with O&G companies in delivering solutions for complex problems such as Offshore O&G pipeline corrosion rate prediction, O&G pipeline corrosion detection, securing data on clouds and designing and implementing Metocean prediction system and bridging upstream and downstream oil and gas businesses through data analytics. Additionally, he is also working on Big Data transmission, security and optimization problems on High Performance Clouds.

Arunava Roy obtained his Ph.D. from the Department of Applied Mathematics, Indian School of Mines, Dhanbad, and presently works as a STaRShip Scientist at SoIT, Monash University Malaysia. Previously he worked in the Department of Industrial and Systems Engineering, National University of Singapore, Singapore – 117576, CorpLab, SUTD, Singapore, and Department of Computer Science, The University of Memphis, TN, USA – 38111.